12:20:02 OCA PAD AMENDMENT - PROJECT HEADER INFORMATION 06/19/89

Active

Project #: E-25-650 Cost share #: E-25-348 Rev #: 1
Center # : R6641-0A0 Center shr #: F6641-0A0 OCA file #:
Work type : RES
Contract#: DMC-8813387 Mod #: Document : GRANT
Prime #: Contract entity: GTRC

Subprojects ? : Y
Main project #:

Project unit: ME Unit code: 02.010.126
Project director(s):
FULTON R E ME (404)894-7409
CRAIG J I AE (404)-

Sponsor/division names: NATL SCIENCE FOUNDATION / GENERAL
Sponsor/division codes: 107 / 000

Award period: 881115 to 891031 (performance) 900131 (reports)

Sponsor amount	New this change	Total to date
Contract value	0.00	37,984.00
Funded	0.00	37,984.00
Cost sharing amount		380.00

Does subcontracting plan apply ?: N

Title: INFORMATION FRAMEWORK TECHNOLOGY FOR INTEGRATED DESIGN/ENGINEERING SYSTEMS

PROJECT ADMINISTRATION DATA

OCA contact: David B. Bridges 894-4820

Sponsor technical contact	Sponsor issuing office
TONY C. WOO	SHIRLEY A. WOODS
(202)357-7676	(202)357-9602
NATIONAL SCIENCE FOUNDATION	NATIONAL SCIENCE FOUNDATION
ENG/DMC	DGC/ENG
WASHINGTON, D.C. 20550	WASHINGTON, D.C. 20550

Security class (U,C,S,TS) : U
Defense priority rating : N/A
Equipment title vests with: Sponsor
NONE PROPOSED

ONR resident rep. is ACO (Y/N): N
NSF supplemental sheet
GIT X

Administrative comments -
REVISION ISSUED TO SET-UP SUBPROJECT E-16-A18 (P.I. IS J.I. CRAIG/AE);
XFER $11,082 TO SUBPROJECT

```
CA                               Maintain Closeout                       09-DEC-1997
CA0140                                                             WS12       [AQ]
+-- Document Header ---------------------------------------------------------------+
| Project Number E-25-650                          Doc Header Id 14085              |
| Project Title INFORMATION FRAMEWORK TECHNOLOGY FOR INT      Status T              |
| Award Period: From 15-NOV-1988  To 31-OCT-1989                                    |
| PDPI FULTON                ROBERT          Sponsor NATL SCIENCE FOUNDATION/GENE  |
| Contract No DMC-8813387                                                           |
| Prime Contract No                                                                 |
| Source Document Header                                                            |
| OCA File No 02.107.000.88.165                      BOA No                         |
| Security Class U           Unclassified                                           |
|                                                                                   |
+-+- Closeout ---------------------------------------------------------------------+-+
  +-----------------------------------------------------------------------------+

Title of the project.
Count: *2          ^                                                    <Replace>
```

E-25-650

APPENDIX V

NATIONAL SCIENCE FOUNDATION Washington, D.C. 20550	**FINAL PROJECT REPORT** **NSF FORM 98A**	
PLEASE READ INSTRUCTIONS ON REVERSE BEFORE COMPLETING		
PART I—PROJECT IDENTIFICATION INFORMATION		
1. Institution and Address Georgia Institute of Technology School of Mechanical Engineering Atlanta, GA 30332-0405	2. NSF Program ENG/DMC 4. Award Period From 11/15/88 To 10/31/89	3. NSF Award Number 5. Cumulative Award Amount $38,364.00
6. Project Title Information Framework Technology for Integrated Design/Engineering Systems		

PART II—SUMMARY OF COMPLETED PROJECT *(FOR PUBLIC USE)*

Over the past two decades there has been a growing concern with the need to use computing technology to manage information associated with the development of complex engineering products. The benefits, issues and potential approaches have been emphasized in a number of initiatives including the NASA sponsored projects IPAD (Integrated Programs for Aerospace-Vehicle Design) and TMIS (Technical Management Information System), the Air Force Sponsored projects ICAM (Intergrated Computer Aided Manufacturing) and IDS Integrated Design Support) and the NIST coordinated national/international efforts of IGES (Initial Graphics Exchange Specifications) and PDES/STEP (Product Data Exchange Specifications). Much of this technology is being incorporated in the DoD Computer-Aided Acquisition and Logistics Support (CALS) strategy. These efforts all recognize the need to use digital technology rather than paper technology to create, document and control the development and support of complex engineering products. What is less clear, however, is the need for extensive research in a broad array of topics to make the paperless world possible. To that end a workshop was convened and its findings are summarized in this report.

The Workshop brought together distinguished representatives from industry, academia and government who are knowledgeable from both a theoretical and applications perspective. The report points out that the "science of engineering computations" has made great strides and achieved many benefits toward the solution of engineering problems but the time has now come to consider a new plateau, the "science of engineering information management". In this new plateau the disciplines interactions are complex, but the benefits great and profound. It is a science whose time has come and government, university, industry partnerships are needed to address this technology of engineering information management. Research across a broad front is critically needed to provide the capability for digital product development in the 21st century. The Workshop participants believe this report provides the outline of a national research agenda in engineering information management and suggests the research components be:

Engineering Product and Process Description
Engineering Information Dynamics and Data Models
Very HIgh Level Languages and User Interface
Engineering Decision Support Systems

PART III—TECHNICAL INFORMATION *(FOR PROGRAM MANAGEMENT USES)*

1. ITEM *(Check appropriate blocks)*	NONE	ATTACHED	PREVIOUSLY FURNISHED	TO BE FURNISHED SEPARATELY TO PROGRAM	
				Check (✓)	Approx. Date
a. Abstracts of Theses					
b. Publication Citations					
c. Data on Scientific Collaborators					
d. Information on Inventions					
e. Technical Description of Project and Results					
f. Other *(specify)*					

2. Principal Investigator/Project Director Name *(Typed)*	3. Principal Investigator/Project Director Signature	4. Date
Dr. Robert E. Fulton		8/17/90

A WORKSHOP ON

INFORMATION FRAMEWORK TECHNOLOGY FOR INTEGRATED DESIGN/ENGINEERING SYSTEMS

BY

Robert E. Fulton
James I. Craig

Sponsored by the

NATIONAL SCIENCE FOUNDATION
and
GEORGIA INSTITUTE OF TECHNOLOGY

Callaway Gardens, Georgia
March 13-15, 1989

A WORKSHOP ON

INFORMATION FRAMEWORK TECHNOLOGY FOR INTEGRATED DESIGN/ENGINEERING SYSTEMS

BY

Robert E. Fulton
James I. Craig

Sponsored by the

NATIONAL SCIENCE FOUNDATION
and
GEORGIA INSTITUTE OF TECHNOLOGY

Callaway Gardens, Georgia
March 13-15, 1989

FORWARD

As we approach the paperless, digital based world of engineering, we are sailing on a new ocean of complicated interdisciplinary interactions. We are defining a new set of semantics and tools for the description and use of digital based engineering activities. The technology to achieve this new order embodies the science of engineering/ manufacturing/support, the art of linguistics, the psychology of human interaction, the logic of mathematics and the preciseness of digital computing. As we approach this world, engineering is embarking on a new method of communication which of necessity must take place through evolution not revolution.

The technology is highly multidisciplinary in ways unfamiliar to most engineers. It must also deal with an existing legacy of todays paper oriented engineering process and yet in 25 years be a capability far advanced from today. The fruits of this technology can lead to dramatic improvements in product quality and capability, reduced development times, reduced cost and improved capabilities for society. Such advances will be necessary to compete in the international market place in the 21st century.

It is a technology that is needed and yet one where neither the problem nor the solution is well defined. It is also an area of immense research opportunity for engineering automation technology. Since the advent of the digital computer in the 1940's, the science of engineering computations has made great strides and achieved many benefits from combining digital technology, mathematics, engineering disciplines and logic to the solution of engineering problems. The time has come to move to the next plateau, the science of engineering information management. The disciplines are more, the interactions more complex, but the benefits greater and more profound. It is a technology whose time has come. Government, university, industry partnerships must join together to address the critical technology of engineering information management. Research across a broad format is critically needed to provide the capability for digital product development in the 21st century. This report provides the outline of a national research agenda.

ROBERT E. FULTON
JAMES I. CRAIG
Workshop Co-Chairmen

EXECUTIVE SUMMARY

Over the past two decades there has been a growing concern with the need to use computing technology to manage information associated with the development of complex engineering products. The benefits, issues and potential approaches have been emphasized in a number of initatives including the NASA sponsored projects IPAD (Integrated Programs for Aerospace-Vehicle Design) and TMIS (Technical Management Information System), the Air Force Sponsored projects ICAM (Intergrated Computer Aided Manufacturing) and IDS Integrated Design Support) and the NIST cordinated national/international efforts of IGES (Initial Graphics Exchange Specifications) and PDES/STEP (Product Data Exchange Specifications). Much of this technology is being incorporated in the DoD CALS strategy summarized in Appendix C. These efforts all recognize the need to use digital technology rather than paper technology to create, document and control the development and support of complex engineering products. What is less clear, however, is the need for extensive research in a broad array of topics to make the paperless world possible. To that end the workshop noted herein was held and this report gives its findings.

The Workshop brought together a distinguished group of experts from industry, government and academia with expertise in engineering and computer science. The group held intensive discussions on the process of product development and support and the relationship of information technology to that process. It identified, discussed and documented many research issues which need to be addressed. The group concluded with a strong recommendation for the establishment of a national research program on engineering information management and suggested that the components include

Engineering Product and Process Description
Engineering Information Dynamics and Data Models
Very High Level Languages and User Interface
Engineering Decision Support Systems

It is strongly believed that this research will require the concerted joint efforts of industry, government and academia and that it will require multidisciplinary teams from such areas as engineering, computer science, social science and mathematics. It is also believed that such research is necessary to provide the basis for effective industrial competition and national security into the 21st century.

TABLE OF CONTENTS

ABSTRACT

In recent years there has been an explosion in the development and use of computer based tools to support engineering, design and manufacturing. In most cases, however, the tools have been developed independently and applications are implemented in a largely autonomous fashion. To improve engineering productivity there is a critical need to advance the technology for management of the information in the analysis, design and manufacturing process. This report summarizes findings of a workshop with industry, university and government representatives to help identify critical research issues in engineering information technology which need solution to support the design of complex engineering products into the twenty-first century.

1. BACKGROUND

The development and practical application of new and innovative technologies to the industrial processes in this country is essential to the improvement in overall productivity and international competitiveness. One of the most promising technologies for achieving large and long-lasting improvements in productivity is the rapidly growing use of automated, computer-based tools for information processing and decision-making.

In recent years there has been a virtual explosion in the development of these tools, especially in the areas of CAE/CAD/CAM (computer-aided engineering, computer-aided design, computer-aided manufacturing). Today, there are numerous software packages available for such areas as geometric modeling, electrical design, structural analysis, thermal systems simulation, and manufacturing process planning. Many of these systems have the equivalent of at least a quarter million lines of code and represent investments by their developers of tens to hundreds of man-years of effort.

The use of such tools in the practice of engineering is becoming almost commonplace in some fields, and some companies have made significant accomplishments in implementing these technologies. In almost all cases, however, the tools have been developed independently, and the applications are implemented in a largely autonomous fashion. The result is all too often a "Tower of Babel" with each application having its own problem definition language and means for representing information, thus making it almost impossible to share this information with cooperating groups. This results in an environment composed of what has been denoted "islands of automation".

It has become increasingly clear that the use of these computer-based tools should be viewed not simply for analysis or simulation but rather more broadly for information processing and decision making using complex and versatile computer systems. From this perspective the concern must be with the processes by which engineering information is created, manipulated, managed, and acted upon throughout the total engineering-manufacturing enterprise.

Today, the capability to manage engineering information has developed in limited and isolated ways with inadequate treatment of the total set of information issues so important to current and future engineering productivity. As a result there is now a critical need to address these issues through the effective advancement of technology for the management of engineering information in the analysis, design and manufacturing processes. An example of a future approach to an information based design system is illustrated in Figures 1.1 and 1.2 Some of the activities that will interact with the information system during a design life cycle are illustrated in Figure 1.3. The motivation is the recognition of a major need for integration of engineering technology across the full set of issues pertinent to its development and use. The key issues are the technology for management of engineering information and the development of a new breed of engineers trained in information management concepts.

While information technology research is underway in many areas, its relevance to industrial productivity is often spurious and, at best, limited. A more organized and clearer understanding of engineering information technology needs is required to guide advanced research in this important area. To respond to this need the National Science Foundation and the Georgia Institute of Technology held a Workshop on "Information Framework Technology for Integrated Design/Engineering Systems" at Callaway Gardens, Georgia on March 13-15, 1989. The workshop brought together key industry, university and government representatives to help identify critical research issues in engineering information technology which need solution to support the design of complex engineering products into the twenty-first century. Participants are noted in Appendix A and the Workshop Agenda is given in Appendix B.

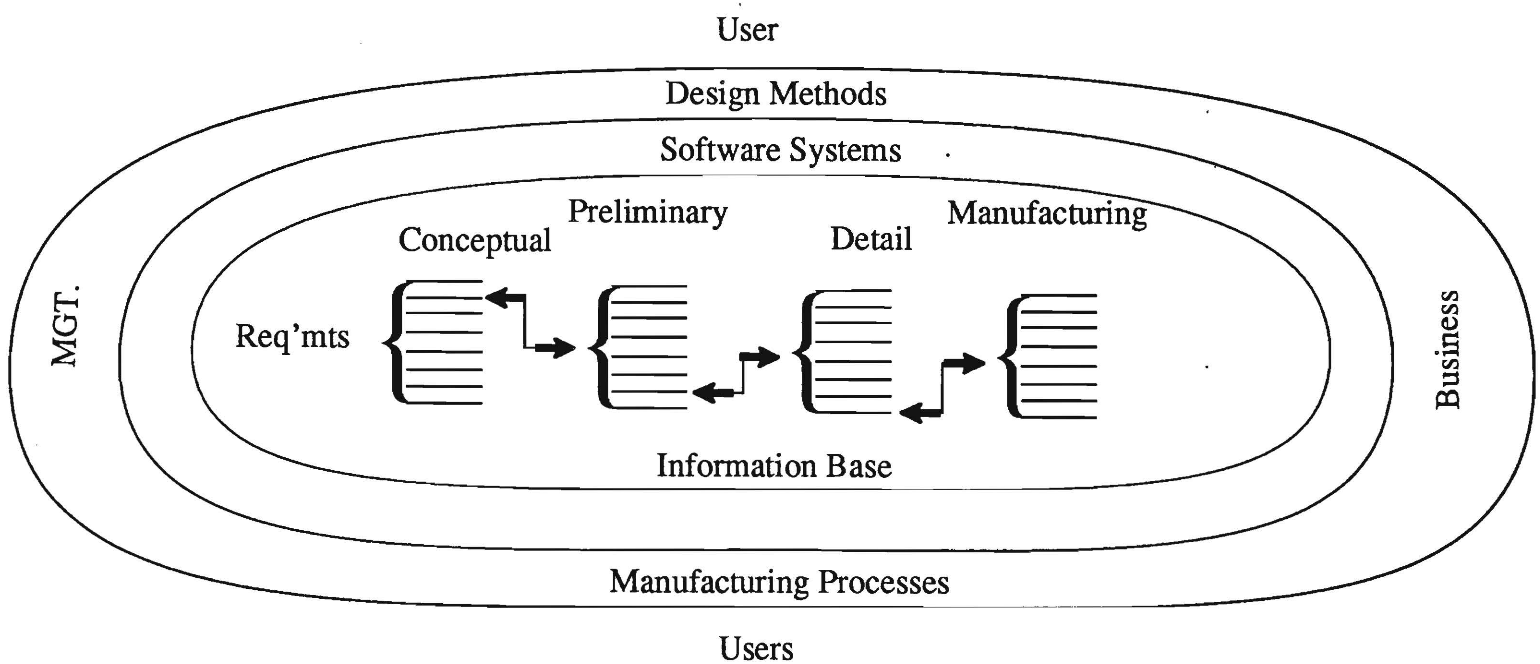


Figure 1.1 Prototype Information System.

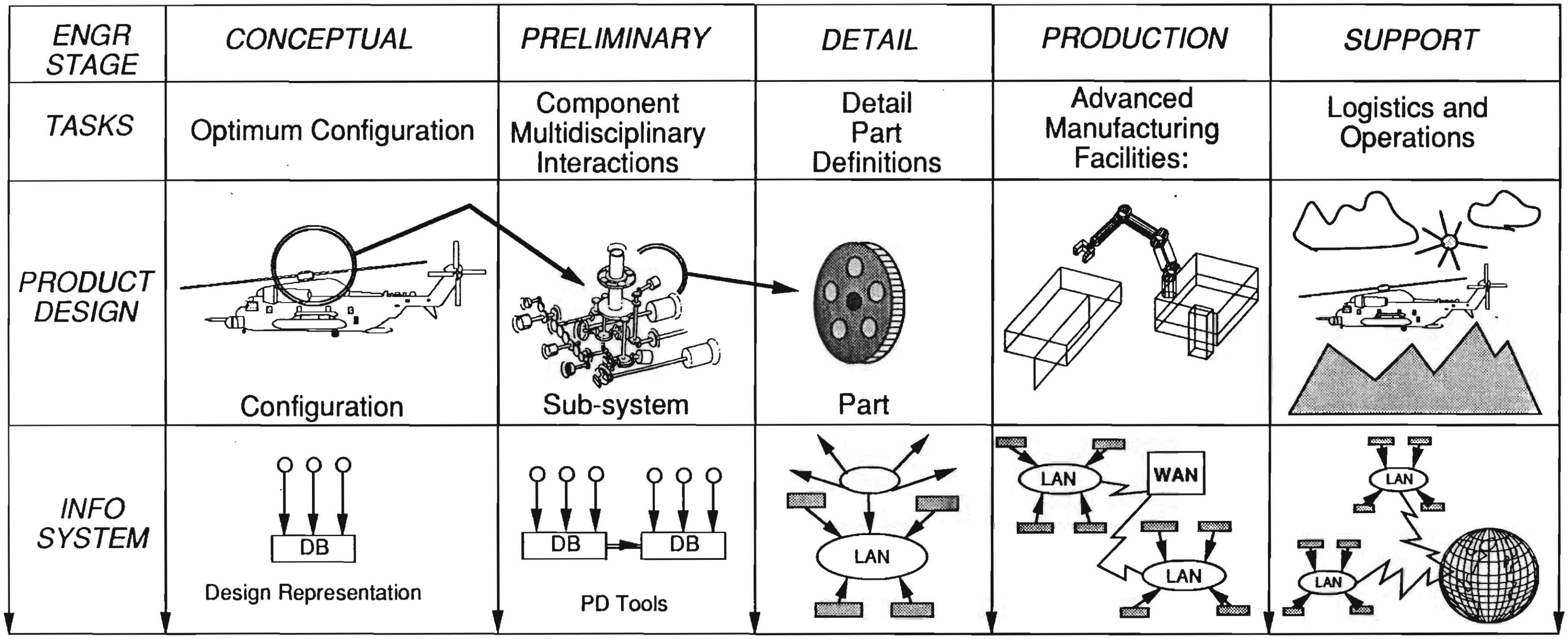


Figure 1.2 Prototype Design Approach.

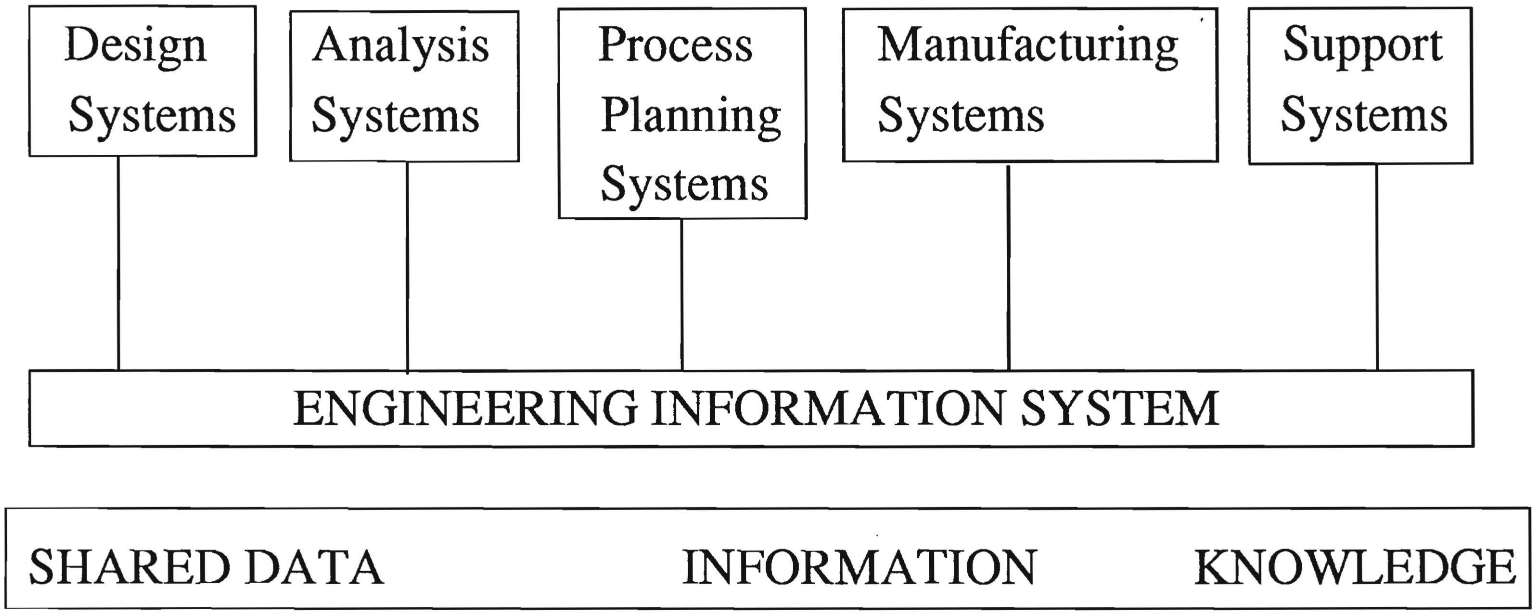


Figure 1.3 Representative Activities Using the Engineering Information System.

2. WORKSHOP OBJECTIVE

The objective of the workshop was to investigate and identify the critical issues in managing engineering information. It specifically addressed the following two critical issues

(1) the key thread of engineering information associated with the design and development of complex engineering products

(2) the access to, uses of, and operation on that data which must be supported to achieve high quality and timely engineering development.

The engineering information base includes many different data types such as geometrical, analytical, mathematical, numerical, as well as features, rules and guidelines. These data types exist at all stages of design, yet some are important baseline information from which other data is derived. There exists a thread of information over time which carries the design definition as decisions are made, evaluations carried out, modifications incorporated and the design matures. Understanding the characteristics of this thread of information and how it is used is a critical issue.

The operations on engineering data are many and varied and include creation, modification, enhancement, extensions, computation, storage, retrieval, integration, interfaces, configuration control and protection. While these operations are often typical data management operations, they take on different dimensions and meanings within the context of supporting the design, construction and operation of complex engineering products when activities such as those in Figure 1.3 regularly use the data. Furthermore, the whole process is highly dynamic; that is, the information and its organizational structure changes and grows by several orders of magnitude in poorly defined ways as a design definition is refined, finalized, manufactured and supported throughout a development and operation process that may cover a decade or more in time. Understanding the special information management needs of engineering data is a critical issue.

To provide a structure for addressing engineering information management needs, the workshop focused on three major areas

(1) Identifying requirements for an engineering information system.

(2) Determining how information is to be structured or organized.

(3) Determining how to design the information base.

Guidelines given to participants relative to addressing the above areas included:

User Interface Needs: The user interface to the information base is especially critical for providing the designer with convenient and readily understandable facilities to interact with the information base, for controlling its development, and for enabling it to serve as an effective extension of the designer's capability. Questions which need addressing include the characteristics, structure and capabilities of the user interface facilities.

Representation of the Topology and Geometry of the Designed Object: Geometry is one of the critical threads through the design process and one of the unique features of engineering information. As a design evolves multiple functional representations of the geometric design description evolve. These typically have different levels of abstraction and serve different purposes. For example, current geometric models typically including such characteristics as two dimensional layouts, three dimensional wire frames, three dimensional solids, and parametric, finite difference and/or finite element models. Each model can produce an extensive data base and the inter-relationship among several models is often complex and nonunique. The interplay between the geometric definition and functions such as flatness, smoothness, roughness and cavities is also poorly defined. Issues that need addressing include the characteristics and approach to achieving a unique and perhaps new topological definition of geometric information for an evolving design which can have different levels of abstraction and support many uses. Other issues include ways to support geometric design information which encompasses functions as a part of the geometric definition.

Product Intent Versus Product Description: The definition of a product includes a wide variety of data including geometry, specifications, manufacturing process, rules, tools, etc. The product intent includes an alternate array of information types. Issues that need addressing include effective and comprehensive ways to define a product, to support its evaluation and production, and to relate those definitions to the product requirements. Other issues are related to the ability to maintain and correlate new product definitions during the early phases of design when several design alternatives are frequently retained.

Informational Dynamics and Data Models: An especially critical characteristic of engineering data is change. Not only does the information content change, but more importantly, the structure and character of the information changes as a design matures throughout the design and production process. Furthermore, when multiple designs for various components of the overall system must be maintained, the inter-relationships and configuration management of data models and data types becomes a significant problem. Issues that need addressing are approaches to define data models for an evolving design, ways to accommodate the steady refinements of design, and means to incorporate effective audit trails.

Interaction of Material Characteristics Informational Structure with Design/Engineering: Designs are concerned with real objects made of different types of materials. The interaction of material alternatives with the design is one of the dominant design interactions for products that will be designed for the 21st century. Material choices will become almost unlimited and the selection of the material will become intimately entwined with the products. In fact, the material will often be tailored to meet a special design need. Material data bases are already under development in such areas as plastics, metals, lubricants and composites of all types. It is unclear whether such data bases will be structured to best support an information-driven design process of the future. Issues to be addressed include the needed material information structure to support design and the ability to deal with a wide variety of material types in ways well suited for engineering design decisions.

Software Utilities for Integrated Design Engineering: A critical issue for the effective computer based design of future products is the software utilities to aid the process. Such utilities include graphics, data base management, user aides, tutorials, expert systems and many others. Issues to be addressed are the adequacy and effectiveness of such aides and areas where significant improvements are warranted.

Support for the Design Decision Process: Design is a continuous decision making process over many or few alternatives and often over long periods of time with many participants. Yet design decision-making is not well understood and two designers often reach different solutions to the same problem or similar solutions via different approaches. The information base must provide the data to serve as the basis for decisions and to store the resulting decisions themselves. As such there must be an intimate relationship between design decision-making and information management. The capabilities of the data base must not dictate in advance the decision making process but must instead support it. Issues that need addressing include how design decisions interact with the data and the flexibility and features required in an information base to support unencumbered, creative design.

To address these and other issues the workshop began with presentations on relevant technical issues after which the participants broke up into four groups under the leaders noted in Appendix B. The groups focused on the following topics taking into account the issues noted above.

-ENGINEERING PRODUCT AND PROCESS DESCRIPTION

-INFORMATION DYNAMICS AND DATA MODELS

-VERY HIGH LEVEL LANGUAGES (VHLL) AND USER INTERFACE CAPABILITIES

-ENGINEERING DECISION SUPPORT SYSTEMS

The following sections summarize some of the issues discussed, views expressed and research priorities identified by the groups in the four focus topics. No attempt has been made to eliminate all natural overlaps and redundances which take place caused by the complex interactions among these topic areas. Rather, the sections reflect the collective views of each individual group related to the topic considered.

3. IMPACT OF THE WORKSHOP

Results from the workshop can provide directions for the National Science Foundation, other agencies, and the industry in general. The following briefly describes the potential impact of the workshop on these organizations.

3.1. Educational Institutions

The cost of training engineers in the industry is expensive. Educational institutions need to take a more active role in training future design engineers in the fundamentals and application of information technology. The workshop can provide a catalyst for these organizations to start thinking about revising their curricula to take advantage of the new methodologies for information based engineering and manufacturing process.

3.2. Funding Agencies

The document resulting from the workshop can provide directions for the funding agencies as to what research needs to be conducted at this stage in order to to facilitate development of new information technologies. It can also provide long range planning and directions for research to continue to study the basic problem of designing efficient and cost-effective systems in a minimum of time.

3.3. Industry

Workshop results can provide directions for both the engineering and computing industry on needed information technology to support future design processes. It can guide engineering companies on approaches to implementing information based design approaches and to computer and software companies on needed software/hardware capabilities to support such processes.

4. ENGINEERING PRODUCT AND PROCESS DESCRIPTION

The "Islands of Automation" in the product development process are increasing in volume at an alarming rate. There is a tendency to address these islands as a basic problem and to focus attention on the hardware/software network architecture on which these islands are built. It is now becoming clear that the information generated and transmitted is the real investments of the islands of automation and this information is the real product of the automated system itself. In fact, without the information to be stored, retrieved, used, sorted, manipulated and transferred, the automated system has no purpose. Therefore, in the design/build/maintenance lifecycle process one must address the "information product" of the automated lifecycle system, in the same manner that one addresses the "manufacturing product" of the development lifecycle system.

4.1 Requirements for Engineering Product and Process Description

In the development of complex engineering products such as automobiles, aircraft, ships, computers and weapons, there are many new requirements for describing the product and process such as

* 100% Digital Product/Process Definition
* Single Source/Entry of Data
* Sharing of Data
* Integrated Customer/Supplier Networks with 24 Hours World Wide Access
* Electronic Ordering, Billing and Payment
* Design to Cost Efforts
* Quality Improvement Programs
* Teaming and Partnerships
* Use of Integrated Design/Build Teams

Each of these bring a unique set of detailed requirements. The computing industry is attempting to address these needs with enabling technologies and new techniques/methods such as

* Data and Process Modeling Techniques and Methods
* Information Exchange Standards
* CASE Methods and Tools
* Object Oriented and Knowledge Based Products

Each of these new technologies add to an already growing base of islands of automation. At the same time industry is overtly or inadvertently making dramatic investment in its technical information. Without an understanding and analysis of the information requirements and the establishment of an integrated approach to addressing the entire scope to these needs, one will only move from one automation island to the next by creating new languages and more deposits of information. Each introduction of a new technology brings about an investment decision on the part of industry. The key decision is whether or when to re-tool the information factory? At the same time this information must be maintained until the product is no longer in service. During the in-service life of the product, one must have access to the information to provide service to customers and to protect the assets of the corporation for audits and warranties. In many cases, the lifespan of the information goes well beyond the lifespan of the software, hardware and network tools used in its initial creation. One must also understand how to manage the introduction of new development technology while protecting the integrity, usability and maintainability of the existing information base.

A key to these issues is the description of the engineering process and product; some facets of understanding this process and product description are

1. Understanding the Information Requirements
    * Who needs the information?
    * How is it used?
    * Where is it created and where is it used?
    * When is it needed?
    * How long must it be available?
2. Configuration Management

* Maintaining the configuration and relationship of all the information used to analyze, define, build, and administer a product.
* Managing the configuration of data and process models used within the design/build/maintenance lifecycle, some of which are typically not under the control of the manufacturing process.

3. Integration of Information to Support all Design Disciplines.
    * Establishing systems to allow the integration of detailing designs into subassemblies and major assemblies.
    * Supporting the establishment of electronic mock-ups of large manufacturing products.
    * Supporting the use of design information generated for one discipline by another discipline.

One concurrent engineering tool is Quality Function Deployment (QFD); but a need exists for a generic modeler for QFD. Figure 4.1 illustrates the current communications of the many islands of automation exchanging data. There is a high probability that these data elements have different structures and formats because they likely have been created or changed at different phases of the product creation. Figure 4.2 illustrates a data sharing environment with a central "Integrated Data Management" function. This function manages and controls the integrity of the data elements and assure that all design functions have consistent data elements. Figure 4.3 illustrates the communications and data sharing environment of the aerospace industry (typical of the USAF ATF program) with three contractors, the USAF System Program Office (SPO) and subcontractors. In this environment there is data sharing within nodes and between nodes. These charts illustrate the key role that data plays in the engineering product description process.

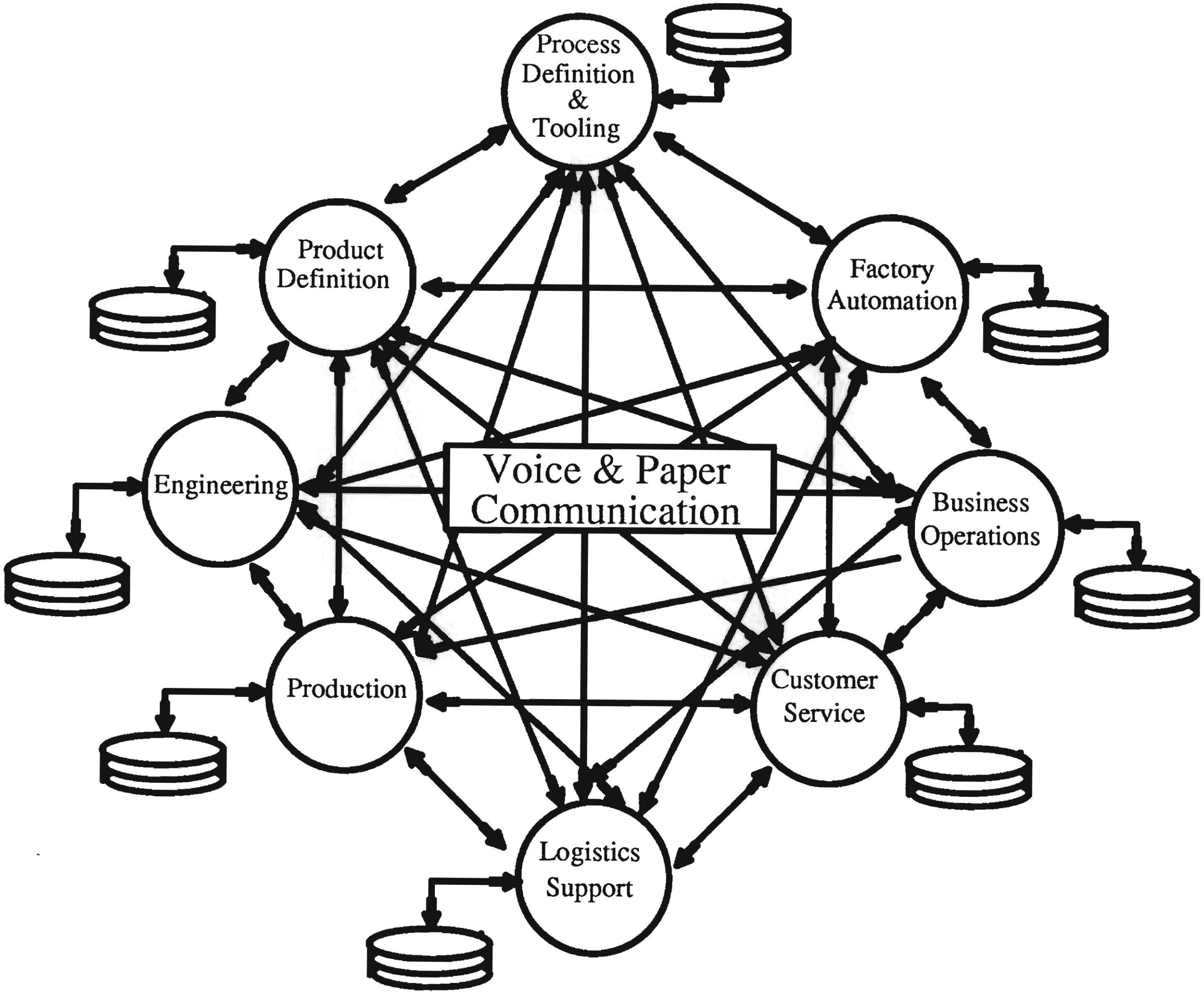


Figure 4.1 COMPUTER AIDED MANUFACTURING

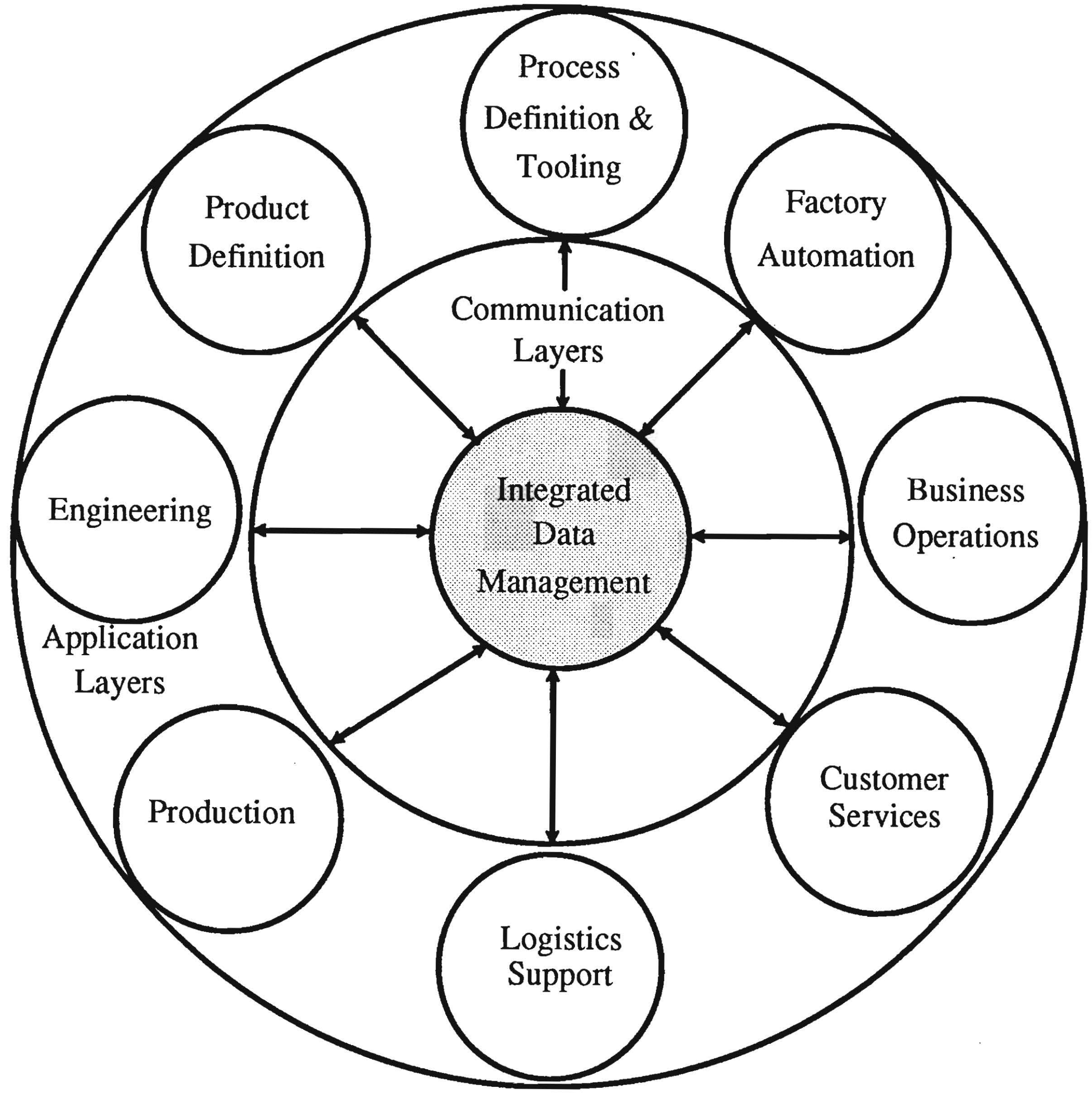


Figure 4.2 TO COMPUTER INTEGRATED MANUFACTURING

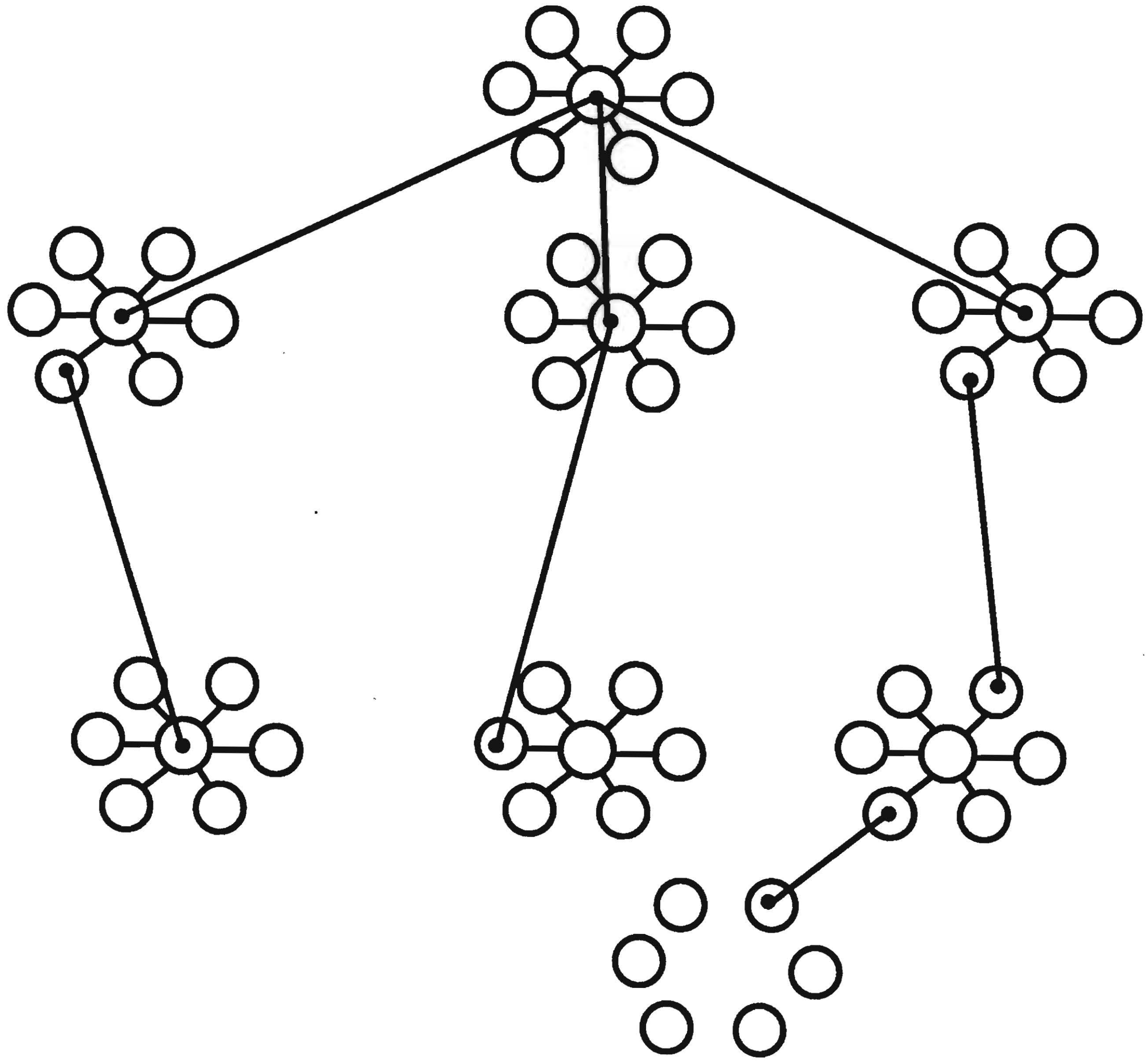


Figure 4.3 TO INTERORGANIZATIONAL CIM

(Multi-Enterprise Production Partnerships)

4.2. Needed Information Technology Capabilities

The automation technologies now permit significant concurrency of engineering and manufacturing through the sharing of data as both the product design and the development process are being created. Facilitating data sharing requires standardized product definition models, manufacturing process models, and support process models. The concept of an "Integrated Product Database" requires that these product and process models and the associated data elements be defined and shared as a common data. The DoD CALS (Computer-Aided Acquisition and Logistics Support) strategy is moving toward requiring an Integrated Product Database for future weaponse systems (Appendix C).

The engineering product and process description area has many needed capabilities in information technology. These can be categorized in terms of product modeling, process modeling, data management and control of product and process modeling, knowledge engineering and acquisition and data communication. The following subsections provide a brief outline of current and needed capabilities in these areas.

4.2.1 Product Model

The Product Data Exchange Specification (PDES) is an emerging standard for the representation of product data. One of the major areas of focus is on the representation of mechanical parts and assemblies. One way to view the product definition is a solid (b-rep) based representation of all the information that would normally appear on a part drawing. This includes such information as geometry, topology, form features and tolerances. While current draft proposal for PDES contains information models for these topical areas, they are by no means complete. Further direction for PDES include the development of the specification to cover a much broader array of applications and the development of sophisticated representation of applications and the development of sophisticated representation of PDES using object oriented or knowledge based representations. For the continued development and evolution of PDES, the group felt research contributions are needed in such areas as the following.

Solids Modeling. Current b-reps have limited geometric curves and surfaces but provide topological constructs to further organization of representations. Non-Uniform Rational B-Splines (NURBS) linked with surface modeling techniques can represent more complicated surfaces but often lack the topological information available in b-reps. Integration of these modeling capabilities needs to explored. Non-Manifold Topology (NMT) has the capability of allowing a uniform representation for wire frame, surfaces and solids. NMT could be a very useful tool for representing some tolerance information such as datum plans, centerlines, etc. It could also be useful for representing conceptual design information. Current NMT modelers have been developed for polygonal representatives but need to be extended to general b-rep or sculptured surfaces. A tolerance model integrated with the solid is still a topic for active research (NSF recently held a workshop to address this area). Also needed are solid modeling tools for manufacturing support in such areas such as N/C code generation, fixturing, and assemblies.

Capture of Design Intent. The PDES activities also indicate (at least for mechanical parts) that capturing the decision and assumptions made during design is just as much a part of the product representation as is the final geometry. Research is needed on ways to incorporate such information into the product design process.

Implementation of PDES. Much of the early work for PDES focuses on active and passive file levels. Object and knowledge based implementations appear desirable, but these new technologies have not yet been demonstrated as feasible. Prototype implementation testing of these and related concepts would be highly beneficial.

4.2.2 Process Model

The process of information engineering is complex and requires not only a clear understanding of the "as-is" enterprise but also the needs which must be satisfied by the "to-be" enterprise. When putting together an information systems architecture the available computer systems, operating systems, DBMSs, and telecommunications networks must be defined in a technology architecture. The functions and structure of the organization determine the application architecture, along with the data and interfaces. Finally, the information engineering product must be costed, prioritized and scheduled. There are a number of CASE tools emerging to support the process of information engineering. Through a series of CASE tool diagrams, the application context, functional decomposition, data flow, entity relationships and actions are defined. From this information a data model is built to serve as the basis for data sharing based on a common data structure.

The data dictionary/directory is the foundation of a successful data model. The management of data and control requires precise definition of each data element. The National Institute for Standards and Technology (NIST) has prepared a data dictionary specification denoted Information Resource Dictionary Specification (IRDS) which has been published as a standard for comment and completion. It can be extended to further definition and could be the foundation for development of future data dictionaries. A Request for Information issued by General Dynamics in 1988 to potential suppliers for an "active" data dictionary/directory received an inadequate response and indicated the commercial market has not yet provided adequate products in this area. These events indicate there is good potential for full data dictionary/directory capability if the right combination of research and technological capabilities can be brought together. In addition, an Air Force Integrated Design System (IDS) project workshop on data dictionaries in December 1988 pointed to a need for more concentrated efforts and research on technical data dictionaries and directories. These activites provide a foundation for needed developments in data dictionary technology.

4.2.3 Data Management and Control of Model and Process Models

The transition to a "paperless" data exchange process will require the establishment of electronic data management and control functions and capabilities. The business of buying and selling information must be defined and specified in contracts with schedules, terms and conditions. Many problems must be dealt with as illustrated by the following questions. When is the data in an Integrated Product Data Base considered final? How is the data validated as meeting the requirements? How is the configuration management of the product and the associated data accomplished? How is the data protected from unauthorized access and/or change? What are the limits of liability for shared data and how are they defined and implemented? In the past nondigital world, these issues were handled by an infrastructure of organizations including: engineering, material, contracts, finance, legal, product support, and master scheduling. These organizations or some new functions, such as Information Resource Management, will need to develop the procedures and capabilities for electronic data management and control especially tailored to support the development of engineering products. Research in this area requires participation of many of the disciplines described above and could be carried out across a university or research institute involving participation from the above organizations.

4.2.4 Knowledge Engineering/Acquisition

The graying of the American workforce, especially in the critical areas of design and engineering, will effectively eliminate much of the knowledge base that these people have developed over their work lifetime. It is imperative that this collective memory be effectively captured, saved and disseminated to the next generation in an expedient manner. This will help assure continuity in the advancement of technology and minimize reinventions of known concepts. Research is needed on methods to capture, store and disseminate this knowledge.

The capture of an experienced designer's or engineer's expertise is typically the province of the knowledge engineer. This person is often an expert in computer science with limited experience in the nuances of the expert's area. Research is needed on how to effectively glean this knowledge. Methodologies and models are needed which can be used across disciplines, but which are robust enough to take into account their differences. The model(s) should allow the efficient and effective transfer of information from designers at the end of their careers (in the form of a "memory dump"), as well as from practicing designers so that they can archive their expertise continually. This model and methodology may take the form of a protocol, but should be expandable and modifiable to take into account different levels of knowledge domains and areas.

The information should be sorted in a format that will allow its retrieval at a latter date by various access machines. This will ensure that the corporate memory will not be lost due to hardware problems. As a result, a robust model needs to be developed which captures the information needed to fully describe an expert's knowledge in a form that will be effectively and efficiently accessed. The model should be expandable to take into account the different sized knowledge domains as well as the different areas of expertise.

As models and methodologies for knowledge acquisition and storage are developed they should be tested in the realistic engineering applications. For example, tests should attempt to capture design information from expert design engineers. After capture and storage, the information should be used by other engineers to produce designs, which should then be checked by the experts from whom the knowledge was obtained. Furthermore testing should be performed in a number of disciplines to determine the breadth and validity of the model. Such research must involve joint teams from industry and academia to both develop, test and validate such concepts.

4.2.5 Data Communications

The engineering product and process data must be transferred among many organizations (industry, government, customer, etc.) for a product to be produced, delivered, supported, and maintained throughout it life. This requires a communications system to move voice, video and data in its various formats between sites in a cost effective, timely and accurate mode. The communication system involves such components as networks, protocols, standards and security.

The communications system must have a network of sufficient bandwidth and rate (capacity) to move complete product data models at a cost lower than by shipment on common air/ground transfer of magnetic media. The communication of a product data model must be able to move geometric (full solid models) data and all other attributes of the model. Techniques of data high speed compression/decompression and/or algorithmic decomposition and reconstruction are areas of potential research to facilitate movement of these very large data sets. Remote access and reading/writing of data models without actual movement of the entire file is another area for potential research. It must be a redundant network with sufficient paths and intelligence to route data from site to site without compromise of integrity.

The data communication system must follow common protocols for sending and receiving data in voice, video or data format. The current practice of TCP/IP protocol appears to be phasing out in favor of GOSIP for government communications in the next five years. Industry appears to be moving toward MAP/TOP Version 3.0 for the near term. Planning for communication using the OSI seven layer (GOSIP for government/industry efforts) protocols is becoming essential as world-wide industrial relationships mature. Standards for information exchange must be developed and accepted by all. The development of all seven layer standards is essential to successful transfer of large data models. The data must be totally secure from any degradation of integrity or compromise (intentional or accidental).

The communications system must present a "user friendly" face to the un-initiated casual user. The system must be designed to recognize the user's profile of access privileges and manage the communications system to provide quick, low cost, and secure data access without guidance from the user. This requires a well organized, integrated array of software capabilities operating behind the user friendly screens. Functions such as windowing, navigational tools, a session manager, ad-hoc reports, pre-stored reports, output support, annotation capabilities, all must work together and appear seamless to the user. Standards for these functions do not exist and will be required in a fully networked, multi-enterprise environment.

4.3. Priorities for Research

In summary, the recommended priorities for research in engineering product and process description are in the following areas

Product models

Process models

Data management and control of process/product models

Knowledge acquisition/engineering

Data communications

"Seamless" information access

Fundamental research in these areas is not a single discipline nor a single organization issue. It will require multidisciplinary and multiorganizational cooperation and it will require advances in computer networks and in standardization of system and people interfaces. It will especially require close cooperation among government, universities and industry.

5. ENGINEERING INFORMATION DYNAMICS AND DATA MODELS

Providing an insight into the complex process of engineering information technology to achieve and maintain a discription of a rapidly evolving product or process requires an array of information technology tools. A particularly critical technology issue is the interplay between data modeling methods and the rapidly evolving product or process definition. Figure 5.1 identifies some of the application needs that information technology must support such as modeling the evolving design/manufacturing data, tracking and controlling the evolution of the production process definition and providing timely characterization of constraints. Figure 5.1 also identifies some of the requirements data modeling tools must satisfy such as effective tools, implementation independent front ends, schema evolutions, data type extensibility and a variety of part definition abstractions.

The facilities to support engineering information dynamics is limited and much of the basic technology is still evolving. Figure 5.2 outlines one view of the status of current information modeling tools. Such tools have many capabilities but are limited to static models of information and to relational database management software. Object technology is just beginning to emerge but is still immature. Only a few baseline approaches for integrating engineering applications data are available as examples and test beds.

The slowness with which these tools are evolving is caused by many factors. Current modeling tools were originally targeted to a much more narrow set of applications and an understanding of information dynamics is still rather primitive. Data management software and modeling products represents a major investment by user organization as well as by vendors and it is difficult to evolve to newer concepts. The generic capabilities needed for engineering are unclear and the applicability of these capabilities to a broader set of fields such as construction, services, and business is not apparent. Basically it is a hard problem with limited underlying theory and yet a area of critical importance to engineering information management. Thus it is a fertile area for research. The following sections discuss specific areas for future research.

APPLICATION NEED	DATA MODELING REQUIREMENTS
Expressing designs at metal level	*Effective and espressive object and*
Modeling design/manufacturing data	*functional models, methods, and tools*
Application specifications	*Spcification languages*
Integration of part forms	*Common underlaying meta models*
	Implementation independent front ends
Evolution of part behavior	*Instance evolution of features,*
	versions, and object migration
Evolution of information system	*Schema evolution, addition of new*
(schema, application, organizations)	*types, schema modifications*
Transaction support	*Long transactions, shared transactions*
	partial consisting, check-in/check-out
Complex objects and geometry	*Data type extensibility*
with multimedia support	
Constraints in integrity and behavior	*Declarative constraint language*
Layered views of application	*Abstractions*

Figure 5.1 INFORMATION DYNAMICS AND DATA MODEL REQUIREMENTS

IDEF1X, ER, Relational DBMS, FORTRAN, IMS, CONDASYL
Scattered Independent Tools
No Servers for Design Data
Use of Files, Commercial DBMSs to Capture Complex Design Semantics
Lack of Integration of Business and Technical Data
No Appropriate Interfaces for Engineers/Designers to Access Data + Meta Data

EMERGING CONCEPTS
Object Models/Languages: Standards (PDES)
0-0 DBMSs
Objects + Functions: IRIS (HP), PROBE (XEROX), FUGUE (EIS)
Frameworks for Integrating Heterogeneous Systems (Atherton, EDA, EIS)

Figure 5.2 CURRENT STATUS OF INFORMATION MODELING TOOLS

5.1 Evaluation of Data Models

The evaluation of a data model with respect to a given problem domain, e.g., manufacturing, design, business, etc., must consider how that model will be utilized. If the model does not capture the necessary information, or if the model cannot be implemented efficiently, then the model is not useful, Thus a major issue in model evaluation is the definition of metrics. It is unclear what the criteria is for judging the "goodness" of a model. Currently, no good metrics exist, and it is unclear whether such metrics should be qualitative or quantitative. Furthermore it is unclear what benchmarks should be used in evaluations. Relational data models or database systems have some benchmarks in the area of business applications; however, these benchmarks typically deal with performance issues, such as response time or throughput. While performance is important in engineering databases, business application benchmarks are inadequate for engineering applications. For example, an engineering design transaction may span days while a business transaction typically takes only a few seconds.

Metrics are also needed to measure modelling power. Little has been done for the relational model since it was developed. This is due in part to the fact that the relational model does not offer a rich semantic modelling capability. The development of metrics for measuring modeling power are critically needed.

Recommended research on method for evaluating of data models include:

Test Suites for Data Models. Develop a set of test applications which can be used to test the modeling capability and completeness of languages and development environments. Such tests should cover a representative set of modeling requirements for design and manufacturing data.

Performance Benchmarks for Engineering Application Development Systems/Data Bases. The performance requirements of engineering databases are radically different from the performance requirements of business databases. The research should lead to benchmarks appropriate for evaluating engineering databases and for comparison and evaluation of database products.

Evaluate Existing Object Systems and Object Databases for Engineering Applications. Utilize the above evaluation metrics on existing database management systems relevant to their value for engineering applications, and determine the areas where further developments are needed (both in modeling extensions and performance improvements).

These research topics will require close collaboration between research organizations and industry groups, to ensure that the evaluations represent practical engineering requirements.

5.2. Tools and Methodologies

Extensive design methodologies exist for non-object oriented languages and databases which provide useful guidelines for both applications and systems design which can facilitate system development to achieve correctness and maintainability. Computer Aided Software Engineering (CASE) tools for non object languages are also emerging to support the design, implementation and verification of software development. Graphical fourth generation language (4GL) tools are emerging to support relational database design through simple-to-use graphical front ends. While these tools only cover a part of the system design process, they improve productivity and increased application correctness. They also make the system development processes accessible to a wider community of programmers and non-expert programmers. Most current CASE and database design tools exist for separate environments; however, they do not effectively support the development of systems which combine complex application functionality integrated together with a complex database schema.

Object systems, however, offer a more natural software design environment for application developers, and tend to improve program correctness, maintainability, and extensibility. Object databases can also alleviate many of the problems of traditional database systems (such as referential integrity), and reduce the pitfalls of database design. However, there can be good object implementations based on bad object implements. An object programming language can be used to develop a system which is correct, maintainable, and extensible, but it can also produce a system which is bug-ridden, incomprehensible, and not extensible. Object languages support good applications design, but cannot enforce it. Therefore, object design guidelines support good concept, design but and methodology is needed which encourage "good" applications design.

Similarly, an object database can resolve many of the problems of traditional database systems, but can still provide opportunities for bad database design. Thus guidelines and methodologies are needed also for object database schema design. Object languages and object database provide richer modeling capabilities than non-object systems, training and methods are needed to ensure that these capabilities are exploited. Furthermore, there are currently no methodologies or tools to support retrofitting existing applications and databases with object functionality.

Research is needed on tools and methodologies which will lead to

1. Guidelines and methodologies for object applications and database design where the design goals include correctness, maintainability, and extensibility/reusability.
2. CASE/4GL tools to support object systems design; such tools should be integrated, provide a single environment for systems design, and include application design and database design. The tools and methodologies should cover the entire design spectrum from conceptual to physical design.
3. Methodologies and tools which include transaction and concurrent control modeling; support design of multi-user applications with the close cooperation and interaction

between users typical of LAN-based design environments; and allow detailed constraint specification with constraint languages.

5.3 Models for Design and Manufacturing

The application of database technology to the design and manufacturing domain is a relatively recent phenomenon with previous applications primarily in support of business activities. Most database applications have used hierarchical, network and relational data model and associated DBMSs that do not capture engineering design data very well. Because of past investments and organizational commitments, many companies have had to force their existing DBMSs to meet design and manufacturing applications. In recent years a strong interest has emerged in object oriented models that seek to incorporate the following features:

a. Data abstraction and encapstation: objects are defined into classes; the behavior of a class is captured in terms of operations or methods; and object classes are organized into type hierarchies.

b. Inheritance and polymorphism: by virture of the type hierarchies, object classes inherit the attributes and methods from their super classes. Polymorphism increases reusability and maintanability, by allowing "generic coding".

c. Complex objects: this provides the ability to define new composite objects from previously defined objects in a nested or hierarchical fashion.

The object oriented family can be divided into models like GEMSTONE that add persistence to the object oriented language SMALLTALK so that objects created in a program can be permanently shared. Another family of object systems originates more from the database area and incorporates object orientation and the above concepts as a part of the data model. A series of such models and their implementations have come about in the recent past: e.g. ORION from MCC, the PDM (Probe Data Model) from CCA-XEROX, IRIS from H-P, FUGUE from EIS, and VBASE from ONTOLOGIC. PDM, IRIS and FUGUE use objects and functions to model data, often referred to as object/function models.

The advantages of the object oriented approach are the following. The models are "natural" and allow a designer to map his or her real domain of interest directly into the database; they are highly extensible since new object types may be added or old ones deleted or modified easily; they are supposed to capture behavior in terms of methods and operations or functions.

Although the object oriented models are powerful, existing implementations suffer from the following;

a. No realistic large scale applications in design and manufacturing exist today that use these models.

b. The current implementation are weak in that they suffer from obvious performance problems in terms of offering reasonable response times to queries or compilation of type definitions.

c. The systems of today do not offer appropriate interfaces for design or manufacturing engineers.

There are efforts under way such as PDES or EXPRESS to come up with a standard object model for describing parts, products, constraints, etc. These efforts are still ongoing and it is unclear whether they will be able to capture the rich semantics of design data in full detail. Another drawback in date modeling efforts in real situations is that they need additional training or education on the part of the engineers/designers. Without it engineers are not able to exploit the modeling power of models, tools and the database systems. There seems to be a growing interest in conceptual modeling in general with the increasing popularity of such methods as IDEFIX, NIAM, and ER and the evolution of commercial software tools to support their use.

Todays object implementations are particularly weak for modeling complex designs, supporting multiple representations of the same design, capturing a variety of system constraints, and dealing with the dynamics of information.

Recommended research in the area of models for design and manufacturing include:

1. Overall system performance of object models.

   Although object/function models or just object oriented models (like ORION or VBASE) have many modeling advantages and features, they will not be used by the community at large unless they provide reasonable system performance. Research is needed in modeling performance and addressing issues of strong efficiency, query optimization, transaction processing strategies, etc. No good metrics exist today for measuring the performance of such systems.

2. Execution models of object oriented databases for design applications.

   Design applications have peculiar requirements in terms of dealing with huge objects; they have transactions that may take place over months of time as well as transactions that are shared among many users. Research on transaction models would involve dialing with check-in/check-out, transactions in the context of reasoning, distributed transactions, long transactions, shared transactions concerning control algorithms, update propagation, etc. Execution models should include a controlled execution of methods, triggers, rules, etc.

3. Support for information modeling dynamics.

   Data models are needed that allow instance evolution in terms of assigning new types to instances and versioning. Schema evolution should make it possible to modify the schema by adding/modifying/deleting existing objects and/or methods and functions. In general, a robust DBMS should deal with system evolution gracefully - involving changes in applications, organizational structure etc. This would give a high degree of data independence. Versioning is an important engineering requirement and significant research is needed on capabilities relevant to changing designs.

4. Support for complex objects.

   Although the object/function models have a fairly rich set of modeling features, complex objects are currently treated as aggregation hierarchies. More work is needed to capture the full functional and structural interplay and the modeling of design requirements in engineering systems. The multiple representation, multiple inheritance problems are not solved. Behavioral specifications and constraint modeling also needs more study.

5.4 Integration

Existing engineering environments generally comprise incompatible tools and data servers linked together in a "Rube Goldberg" fashion to provide the semblance of integration. They rely on multiple translators, ad hoc methods of passing data between tools, and manual methods of ensuring consistency of the results. Often this requires the assistance of an expert for each tool, database, or hardware platform to provide expertise in using that component. Incompatibilities exist in data models, programming languages, hardware platforms, underlying data representations, and user interfaces.

Information interchange between engineering organizations is similarly hampered because of external incompatibilities between the tools and databases of the organizations. Although some syntactic problems are being addressed by standardization efforts, sematic incompatibilities remain. The ability to interchange information is based on agreements between the participants. Currently, there is no common representation of engineering objects, particularly of representations of behavior, nor is there agreement on the meanings of engineering objects or operations.

But, engineering organizations often have large investments in facilities hardware, user training and databases. In addition they have cultures and long-standing methodologies that would be costly or infeasible to change. The solution to the integration problem must preserve these investments as well as be extendable to incorporate new components. There is no guarantee that new components will be compatible with each other or with existing tools and databases.

The current lack of integration among engineering tools and databases is discussed in Section 5.2. Some efforts are addressing this situation, notably frameworks to integrate engineering tools being developed by EDA and Atherton Technologies and earlier efforts to integrate heterogeneous databases at INRIA, CCA, and Honeywell. The Air Force EIS project is also attempting to produce a framework that integrates engineering and administrative tools, DBMSs and file systems.

There are many deficiencies in existing models to support integration. For example, approaches that integrate existing tools and databases or extend environments to incorporate new tools and databases by modifying the tools or data are too costly. They are likely to result in a loss of vendor support for the components and the introduction of errors into the software or databases. What is needed are frameworks that integrate components without compromising their autonomy. It is believed that only "federated" systems in which the component tools and databases are left intact are practical for large engineering organizations.

The frameworks must also provide both conceptual models and integrating services. The conceptual models provide data and execution models for the federated systems that describe the structure and behavior of the components and of the federated system itself. They provide a homogeneous layer over heterogeneous underlying components. Integrating services must provide execution control (i.e., procedure invocation and management, transaction management) and facilities to tailor the resulting systems to the needs of specific organizations.

Recommended research in modeling methodology to support integration of heterogeneous engineering data modeling tools and databases include:

1. Conceptual Models. Research is needed to develop powerful meta-models for describing heterogeneous components of federated systems for engineering. They must describe not only engineering and administrative data but also engineering and management processes or procedures that comprise the resulting integrated systems.

2. Execution Models. This research would provide execution paradigms to support execution in a distributed, heterogeneous environment. The execution models must

provide for parallel processing in support of collaborative transactions, as well as traditional database transactions. Moreover the research would result in optimization strategies over the heterogeneous implementations that make up the federated system.

3. Integrating Services. This research would provide the design and prototype implementations of services to integrate heterogeneous software components and databases. Needed services include both mappings of requests for data access to the underlying database servers and file systems and execution services that include procedure invocation and execution control (including transaction management) in distributed environments.

4. Performance Measurement and Enhancement. This research would provide metrics for measuring performance of federated systems to support engineering and extensible optimizers to improve performance where operations are implemented by heterogeneous procedures and data servers.

5.5 Theoretical Issues for Object, Semantic and Conceptual Models

The attempt to formalize complex engineering design tasks brings into focus some limitations of these most important system objects of all, the people themselves. Human conceptual capabilities, their structure and function, form the background for all of the subsequent discussion of "models" in this section. This assumption in turn implies that the disciplines of psychology, philosophy, linguistics, and mathematics are equal contributors to the creation, application, and understanding of models. Consciously held models encode knowledge that allow one to interpret, act upon, and predict interaction with the environment in a purposeful manner. From this view point, a model is a tool, no less than a hammer or a surgeon's scapel.

Along these same lines, a model can serve to tear apart or more precisely specify what was previously comflated. An example of this from the database world are the two abstraction: aggregation and generalization. With these two abstractions, an analyst can distinguish two different kinds of object interconnections; without them it is difficult to express this distinction. Current problems and ambiguities in engineering environments involve "abstractions", "defined type ambiguity", "semantic equivalences", "logical expressiveness and completeness", as well as the more mundane expressions such as "are we talking about the same features?" These concepts reflect an increasing awareness that linguistic, philosophic, and psychological issues are coextensive with mathematical ones.

Given these perceptions, it seems helpful to approach the analysis of engineering models, in particular objected oriented models, from a linguistic, psychologic, philosophic and mathematical perspective. Furthermore, it seems useful to characterize different classes of these object oriented models as kinds of languages. From this perspective it is appropriate to address data models from the perspective of lexion, syntax and semantics; in short, the grammar of the language.

Recommended research on theoretical issues for object, semantics and conceptual engineering data models include:

1. Engineering Oriented Modeling Language. There is a need for modeling languages that are more closely matched with the way human beings interpret their interaction with the environment.

2. Multiple Level Modeling Languages. This means that there is a need for levels of modeling languages that smoothly map from one to the other and are bounded above by natural language. In particular there is a need for a richer logic that can express more of the conceptual requirements than is provided, for example, by first order logic.
3. Formalized Information Modeling Design Theory. There is a need for a formal treatment of object/functional models from both a mathematical and linquistic viewpoint. There is no design theory to go with object oriented models and systems today. There is a need to develop and test such a theory against practical situations.

5.6 Priorities For Research

In summary, the priorities for research in engineering information dynamics and data models are:

Evaluation of Data Models

Tools and Methodologies

Models for Design and Manufacturing Data

Integration

Theory

It requires data modeling tools and evaluation approach, development of models specifically related to design and manufacturing and developing a new theoretical base for modeling technology.

6. VERY HIGH LEVEL LANGUAGES AND USER INTERFACES CAPABILITIES

This area focused on interface issues, ranging from the top user level down into the model and the interfaces between models. In addition, models exist at various levels of abstraction, both because of the requirements for data at various levels of aggregation by evaluation and analysis programs, and because of the needs of designers as they move from highly abstract general goals to specific properties of the final solution.

Thus this topic includes two orthogonal axes:

- between user and model level interfaces
- over the levels of abstraction used throughout the engineering life cycle

Thus tasks that are done by human actors today may be done by computational processes in the future and more trade-offs of this sort will occur in the future. (Will a person design it or a machine?) Thus, the more generic issue of interfaces seemed appropriate.

6.1. Representation Issues When Modeling Geometric and Physical Processes

The present objective of computerized support of engineering design and product life cycle evaluation requires rethinking existing models and abstractions. Current islands of expertise must be extended into a continent of systematic and comprehensive knowledge of the many interactions between geometric and physical characteristics and phenomena, so that one can implement the needed continual product analyses and evaluation. Engineers are very good at details, but often lack critical knowledge for integrating those details systematically.

Understanding the various interactions between geometric and physical problem aspects has a representational side and an infrastructure aspect. The former deals with conceptual or concrete data, whereas the latter builds a network of problem solution methods at various levels of abstraction.

On the representation side, one finds that, for isolated aspects, specific methodologies have been developed. These include shape representation schemas for the geometry of objects, and various abstract or concrete representations for physical aspects such as dynamics, stress/strain, heat flow, etc. Typically, representations flow from a conceptual model of such an aspect and a body of algorithmic solution techniques for evaluation the model. Focus is needed on the representational issues because an integration of different models into a comprehensive suite of analysis and evaluation programs requires interfacing different schemas. An understanding is needed of

1. Whether possible interaction arises between, e.g., heat flow and stress analysis, and how the interaction is accounted for in the representation.
2. If there exists a more inclusive representation useful for more than one problem aspect, perhaps even a universal one.

Since different phenomena have been studied an specialization, their possible interaction is an unknown, and its manifestation as convenient representations needs to be understood before we can make progress. Very little research has been done up to now, but incremental efforts to integrate separate areas of expertise are appearing across the country.

Principal research issues include the following:

1. Geometry - physics interaction
   -mesh generation (e.g., features-based)
   -compliant motion, temporary contact
   -physics from geometry (e.g., collision)
2. Physics - physics interaction
   -interaction of physical phenomena (e.g., wing deflection vs airflow)
   -likely to span many engineering disciplines
3. Shape variation, tolerances
   -statistical distribution, e.g., due to mechanical processes
   -impact of functionality on permitted tolerance
   -accuracy constraints

4. Qualitative parameter value changes

   -bifurcation/catastrophe points when varying parameters

   -constraint derivation

6.2. Evolution of Models and Design Environments

Engineering models and the systems which create them usually assume that the world around them isn't changing, limiting the longevity of both models and systems.

How can engineering models and their support environments be made to allow adaptation and evaluation for addressing unanticipated fundamental changes in techniques, technologies, and problem statements?

Some of the needs which require further investigation include development of techniques to:

1. Merge old and new concepts
2. Merge or add new functions into existing support environments
3. Allow users to dynamically extend database schemas
4. Allow integration of previously unrelated databases, handling both address space and schema identification issues
5. Allow merging of selected portions of models from alternative development paths, where the schemas are identical but data content dependencies exist
6. Allow high level specialization of tools to provide translation between different data base schemas

6.3. Abstraction, Refinement and Representation

Design and evaluation is done at varying levels of abstraction, including conceptual design, preliminary design, detailed design, maintenance/life cycle considerations. Different levels of abstraction should be supported by corresponding data structures that permit various analysis tasks to be performed, as well as by tools for refining the data when going to greater detail, or abstracting it when falling back to earlier/higher conceptual design levels.

The levels of abstraction are not necessarily hierarchical. In fact, different abstractions may coexist and overlap, e.g., considering machining a part, or assembling it, or painting it, and so on. The interaction of different views, and the correlation between varying levels of abstraction, need to be investigated. Based on the interaction, strategies for maintaining consistency need to be developed, and suitable primitives and operations must be designed for abstracting, refining, and modifying objects.

The principal research issues are identified as:

1. Overlapping views of data
   -consistent manipulation, view coherence
   -feature vocabulary
   -feature overlap and interaction
2. Managing abstraction/refinement
   -correlating levels of abstraction and refinement
   -consistency of different levels
   -integration of geometric model representations
3. Primitives at very high level
   -conceptual primitives for design
   -conceptual design definition operations
   -conceptual design evaluation/analysis operations

6.4. User Aspects of Interfaces

This area concerns all of the ways that engineers will interact with the engineering system. Today's systems are generally limited to interactive changes to geometry and to using programming languages, such as FORTRAN or C, albeit with a macro capability, to building parameterized geometric models. However, the engineering process deals with a significant amount of nongeometric data as well. Future systems will require powerful capabilities to describe abstract models that include all those factors that determine the geometry, as well as a variety of means to display and interact with these factors as well as the geometry. This two-way operation would be a significant improvement over today's primarily one-way visualization of only geometry.

The principal user interface research topics can be summarized as follows.

1. Visualization techniques for nongeometrical engineering information

   -Relationships and dependencies among parts

   -Organization and display of large numbers of important attributes

   -Display techniques for comparative analysis of alternatives

   -More easily comprehended display of analytical results

   -Visualization of the design with varying degrees of abstraction and detail

   -How can a potential design space be visualized and searched?

2. New ways of searching, organizing and manipulating engineering data bases. SQL is limited in applicability and requires significant knowledge of the database organization to structure a query.

   -Need an interactive means of experimenting with the search such as a query-calculation capability

   -Engineering models will need to have direct links to data bases of parts, materials, reliability data, cost estimations, etc. How are these links to be expressed and controlled in the data model?

3. Multiple domain-specific views of engineering abstractions.

   -How can a single unified, integrated model be described and built to satisfy the needs of different engineering disciplines involved in a complex design?

   -How can that model be presented and manipulated in a different way by each of the disciplines?

4. Intelligent integrated design environments

   -If the design process is viewed as a compiler or synthesis process, how can environments be built to allow tracing, inspection, and correction of the processes?

   -How can the state of the process be called backward and forward to incorporate changes?

5. Exploitation of semantics in feature-based design. If high level abstractions such as features are used in a design, how can analytical results be tied directly back to the features in a semantic way?
6. Reflection of instance editing in abstract models. How can interactive editing of an evaluated model be reflected back in the abstract unevaluated model?
7. User customizable and extensible interfaces. How can user interface tool kits be constructed to allow for each user customization and extension?
8. A high level language for abstractions, specifications and constraints.
   -Can a high level language be developed to fully describe abstract engineering models including functional specifications and constraints?
   -Can this language be compiled?
9. Comparative analysis of culturally different engineering processes. Research should be done to compare the fundamentally different ways the engineering process is accomplished in different cultures. Do they differ in efficiency, management of complexity, quality of output, adaptation to change, etc.?

6.5. Integrity and Constraint Management

Constraints are a mechanism for representation of knowledge; integrity is a means for dealing with constraints and other forms of knowledge that can be made internal to the model. Constraints can be applied to manage both definitions (e.g., Ohm's Law) and physical laws to design intentions. They are a powerful means to embed knowledge in a model. Constraints must be structured if they are to be managed, otherwise they create a "spaghetti mess". How to partition or structure them is an important issue to their use. Integrity pertains to constraints and other forms of evaluations with well-defined criteria. Violation of integrity is required during design operations. In the end, however, integrity over the total specification is the task of a design effort.

The identified research issues identified for integrity and constraint management can be summarized as follows:

1. Defining constraints

-high level languages

-qualitative constraints

2. Managing constraint updates

   -localization of management

   -temporary release of constraints

   -transactions

   -maintenance across multiple views

3. Partitioning constraints in distributed databases
4. Tracking validity of data based on source and degree of verification
5. Configuration control

6.6. Organizational Interfaces

Information flows between individuals and processes in an engineering environment. Concepts to improve and innovate both the structure and content of this communication could greatly increase the efficiency and flexibility of engineering organizations. This suggests several areas worthy of further study including:

1. Development of systems to model information flow in organizations, including vertical and horizontal communications, information classes and functions
2. Mechanisms for information control and access
3. Techniques to facilitate collaborative efforts and cooperative engineering with two or more entities. These techniques can extend from human protocol issues all the way down to computer network protocols conducive to multiple workstation information dissemination and control.

6.7. Concurrent Engineering and Conflicts

This life cycle of a product involves many design tasks. For instance these could include:

-design of the product itself

-design of the manufacturing processes

-design of maintenance and product support processes

The purpose of concurrent engineering is to determine which of these tasks will be dealt with up-front, during the conceptual stage of the products' life cycle. The intent is to insure that the selected design tasks will be performed particularly well. A key issue is conflicts among these tasks. If they are all to be successful, there must be few if any conflicts among them. For instance, the design of the product itself must not make the product unmanufacturable. Research is needed in concurrent engineering in such areas as

1. Strategies for
   -decomposing life cycle design activity into discrete tasks or projects
   -imposing partial ordering on tasks
   -scheduling task execution
   -promoting parallel conceptual development across tasks and disciplines
   -promoting collaboration, coordination and efficiency my mechanisms such as incremental release of information
2. Methods for identifying potential conflicts among tasks, especially in the conceptual stage of the life cycle.
3. Methods for resolving conflicts
   -before they happen (get it right the first time...)
   -iteratively (after they have occurred)
4. Methods for assessing the risk that hitherto unidentified or unresolved conflicts will cause downstream in the engineering process.

6.8. Priorities for Research

In summary the priorities for research in very high level languages and user interface capabilities are:

- representation issues when modeling geometric and physical process
- Evolution of models and design environments
- Abstraction, refinement and representation
- User aspects of interfaces

- Integrity and constraint management
- Organizational interfaces
- Concurrent engineering and conflicts

It requires high level languages and user interface capabilities specifically tailored to meet the needs of design, manufacturing and support of engineering products.

7. ENGINEERING DECISION SUPPORT SYSTEMS

In the area of engineering decision support systems, a broad range of research issues were voiced in the workshop discussions. Although a consensus could be found on research areas, issues and questions, the issues and questions were not limited to Information Framework Technology (IFT). Research issues more directly concerned with IFT are discussed in the main body of the chapter. A broader range of issues raised by the participants is included as an Appendix D. Fundamental questions concerning the design process and computing environments for design were a recurring theme during the workshop, and this is reflected in the research issues, as well as in the Appendix D:

"What information is required for engineering decision support ?"

"How do we represent the design decision process ?"

"What constitutes a design decision ?"

Many of these questions seem to relate only peripherally to information framework technology. The prevalence of these broadly posed questions, and the lack of definite focus on research issues that are specifically and clearly relevant to information framework technology, are indicative of more fundamental issues.

Thus research in the application of computing technology to engineering design, in general, and in information framework technology, in particular, must be considered in the broader context of research on the engineering design process. Fundamental research on engineering design processes must be pursued first, to pose questions which may be answered, in part, by information framework technology. Thus the answers to these questions do not come from information framework technology alone, but from the relationship between the information framework and the flexible computing environments which can be used to create such a framework. To better understand the decision support reserch issues, the following section outlines the relationship of decision support to other applications of IFT to engineering design, and characterizes the design process.

7.1 Relationship of Engineering Decision Support to Other Applications of Information Framework Technology to Engineering Design

Engineering design can be viewed as a decision making process. For this reason, *engineering decision support* touches on all aspects of information framework technology for engineering design. Design engineers invent alternative design concepts (including manufacturing approaches) and communicate them to other product development team members by creating a partial *engineering product and process description.* The choices implied by these alternative design concepts are then narrowed. Engineering analysis is used to evaluate the design alternatives. In a complex design problem, design decisions are closely coupled, so a convergent design process must be found and followed. This process of inventing, describing, evaluating, and selecting design alternatives defines the *engineering information dynamics* and sets requirements for *data models.* Whether computing technology helps to make engineering decision makers more effective, or not, is determined by the ease with which design information can be created and manipulated in the computing environment. This kind of accessibility depends on the sophistication of *very high level languages and user interfaces* provided by the design computing environment.

7.2 Characterizations of the Design Process

The role of information framework technology in design decision support is closely tied to the nature of the design process and to the role of the computing environment in the design process. Two complementary views of the design process were defined during the workshop. One view considers the design process as a sequence of iterations between requirements and decision-making. (Figure 7.1)

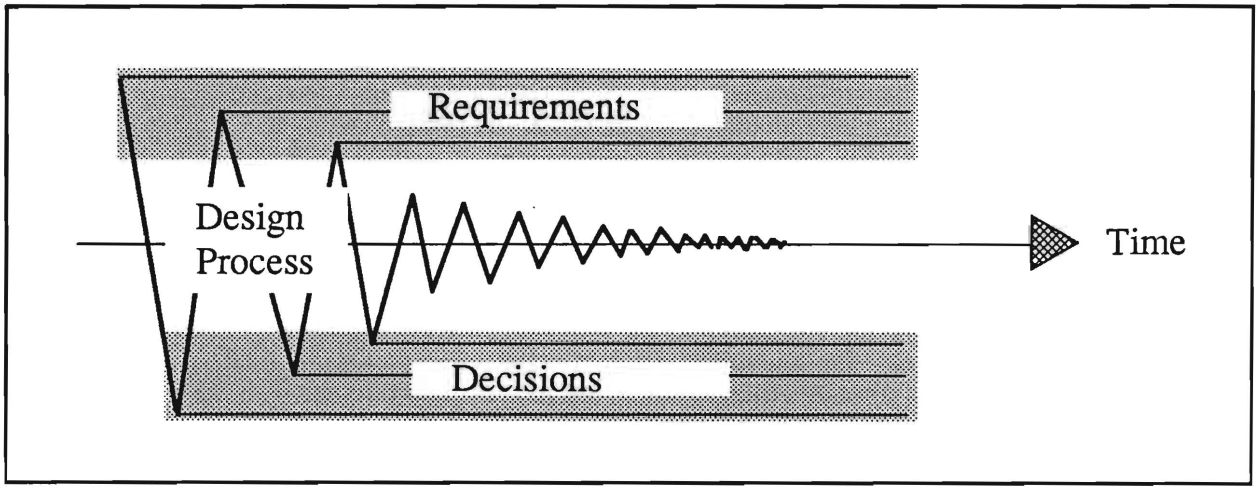


Figure 7.1. The Design Process Involves Iteration Between Requirements and Decisions.

Another view of the design process (Figure 7.2) emphasizes the role of design decisions, application programs, and heuristics in executing state transitions that act on the design state as represented by a design description. The design description includes both product and process models of the design concept. These state transformations successively refine and add definition to the information contained in the design description.

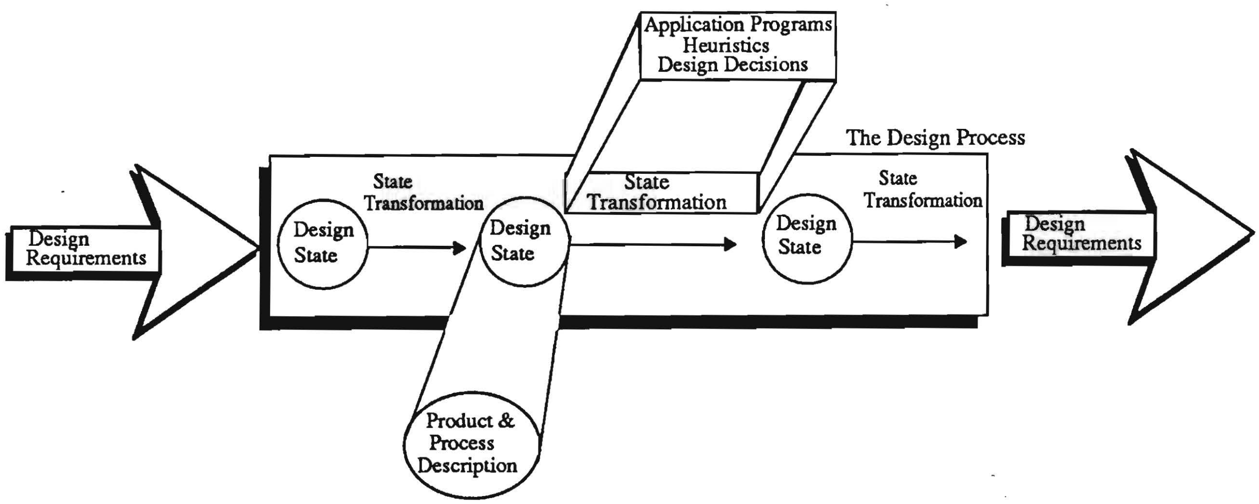


Figure 7.2. State Transformation View of the Design Process.

7.3 Research Issues and Priorities

The tools provided by information framework technology include systems for capturing, versioning, storing, retrieving, and controlling access to data (databases). The integration of processes or procedures (such as "application" computer programs) into the information framework is also important for engineering design decision support systems. Information framework technology can impact engineering decision support systems through four principal areas: computing environment, standards, control and conflict resolution, and user interfaces.

When applying computing technology to design applications, the distinction between data and procedure is highly context-sensitive. Thus, in the discussion that follows in this section, database and process issues are integrated into the computing environment. Research issues bearing on standards are then presented. Design decisions must be made before design data can be delivered, and the information framework should enforce this discipline by effecting control of the design process through a mechanism for conflict resolution. Thus, research issues in the closely related areas of control and conflict resolution are discussed together. Finally, the issue of user interfaces is discussed.

Once research issues have been identified, a number of specific research questions can be posed. These fall into two principal classes: questions leading to a definition of the information content of a "design decision", and questions that might arise in an investigation into the representation of design decision information in an information framework. These questions raise research issues concerning engineering design processes that are beyond the scope of IFT. For this reason, the discussion of these research questions has been placed in Appendix D.

Questions of definition and representation of design decisions in a computing environment must be addressed before an evaluation of the suitability of various technologies (relational, object-centered, etc.) for information frameworks as a means for supporting design decisions can be taken up. It is recommended that this basic research be prioritized at the same level as research into product/process representation. The basis for this recommendation is that a clarification of the interaction of product/process representation technologies with the decision-making process is needed to keep the development of these technologies from focusing on representing decision results to the exclusion of decision alternatives.

7.3.1 Computing Environment

In design applications of computing technology, the distinction between data and procedure is highly context-sensitive. Thus database and process issues are discussed in the context of a *computing environment.*

Database applications currently include relational, object, and file models. Access is through query languages (such as SQL) or operating systems. Hypertext includes several concepts particularly relevant to engineering decision support. Processes and application programs currently being used include design process capture (at least at the level of keystroke memorization in a CAD environment), various engineering analysis tools, design optimization, and knowledge-based design tools (such as finite-element analysis guides).

In databases three aspects emerge that impact the decision support environment. The first is database models. The state of the art in models is assumed to include what is readily available on the market today: relational, object oriented, and files. The second aspect is database access. For relational data models, the standard access mechanism is SQL, and for files the standard access mechanism is the computer's operating system. Object-oriented data models are just beginning to become widely available, and there is no accepted standard access mechanism.

The third aspect of database technology is the emerging area of hypertext and its relation to traditional DBMS issues. At present there are a growing number of hypertext systems that are capable of developing hypermedia that include text, graphics and other elements of information. It seems quite likely that this technology can be used to store text, graphics, and other information that would clearly support engineering decision-making.

In order to implement global databases for decision support, a 3-schema database is required to map the overall conceptual semantic model into application specific external views. In addition, new data types (with either zero, one, or n different domains) must be superposed (or semantically loaded) onto existing schemas. For example, a "weight" could be defined as a "decision-variable", or possibly more precisely defined as a class of "structural decision variables". The database must support decision support system mapping over existing (legacy) data environments and the collection and propagation of the data to the decision points. In a heterogeneous database with local schemas, a mapping must be defined that identifies decision variables that must be collected into decision objects that can be evaluated by an evaluation function.

A design process is an occurrence of something that changes the information content of the database. In order to support the decision support process several stages must be considered. First, an audit trail of the decision-making process itself must be captured. This capability is currently limited to keystroke or function memorization. That is, the computer can be used to record a series of decision-making steps and to play them back much like a text editing computer program can remember a previous sequence of editing commands. At a later stage, the decision-making process must function with whatever standard application software exists in the domain of interest. For example, finite-element analysis and optimization software is the state-of-the-art in structural analyses. Finally, knowledge-based tools that aid the designer in the use of software should be considered as supporting the decision-making process. An example is finite element advising software that can aid the user in selecting an analysis algorithm, element types, etc.. to match a particular problem type.

7.3.2 Standards

Standards currently exist for graphics and exchange formats, but are clearly not yet developed for design decisions. The value of a framework for capturing and managing design decision information within an organization seems clear enough. However, documentation of design decisions, as opposed to product definition, is not usually deliverable under engineering contracts. Most organizations would protect this design decision information as highly proprietary.

The value of standards for exchanging design decision information is less clear, and such standards would be quite difficult to establish. Once in place, they may tend to channel further development along unreasonably narrow lines. On the other hand, it seems evident that the design rationale can be reconstructed from a comprehensive design decision history. The design rationale is invaluable to technical decision-makers throughout the product life cycle. However, equivalent information is available in a comprehensive functional description which should be included in the product/process model. Thus, we are not yet in a position to resolve the matter of exchange standards for design decisions.

The more fundamental question of "how to represent a 'design decision' in an information framework" must be answered first. A completely satisfactory definition of a "design decision" has not been established. The distinction can be drawn between the results of design decisions, which are represented as product/process definition information, and the design decisions themselves. For example, the concept of state transformation should be reflected in the definition of a design decision, but is not needed to describe the results of a design decision.

Currently emerging standards for exchanging product/process descriptions seem to emphasize the representation of these decision results. From the point of view of decision support, representation of the design alternatives that exist before a state transformation is effected by making a design decision is equally (perhaps more) important. In particular, complex relationships exist among attributes of the design alternatives. This complexity is not present in the design description resulting from the decision-making process. In fact, one view of successful engineering design is that wise choices are made precisely to control this complexity, resulting in a design that is simple to use and understand.

7.3.3 Control and Conflict Resolution

The functions of design process control are closely linked to the process of conflict resolution - determination of alternatives and choosing the best from global and local considerations. Design decisions must be made before design data can be delivered. The information framework should enforce this discipline by effecting control of the design process through the mechanism for conflict resolution.

Control of the design process today is primarily implemented through human and paper-based communications, although configuration, version, and password controls are built into CAD tools. Conflict resolution today is primarily accomplished by human decision-makers. However, examples of conflict resolution, or at least, conflict identification tools currently in use include I/O routers (electronic systems design), spell checkers (word processing), and interference checkers (CAD).

The key research issues are then: what information framework technologies are appropriate for flexible implementation of design process controllers, and in addition, how can control and conflict resolution processes be modelled in an information technology framework?

The functions of design process control include:

- identify/collect data needed to make a decision
- propagate changes through database
- identify data affected by a decision

The decision support system must have a control point process that is periodically given control to determine the recommendation or actual scheduling of the next process step. This means that users, robots, etc. are authorized to perform activities of only finite duration, after which the Controller must be invoked to authorize the initiation of a next process step (or set of parallel process steps).

The controller must check the state of pending decision objects (objects that are collections of decision variables obtained from many sources) and if sufficient information has accumulated, schedule an evaluation function to process the object. The output of the evaluator then will generate inputs to other decision objects, which may cause other non-decision processes to be scheduled (or humans to be authorized to engage in various design activities).

Typically conflicts, loops, and cycles will exist in this process. The conflict resolver must detect conflict and must access the various decision objects, re-create (regenerate) the activity sets that generated the conflict, and present the information to a resolver function (or human) for resolution (adjustment).

Conflicts are divided into two categories. The first is identification and the second resolution. With respect to identification, there are certain technologies that are appropriate. These include spelling checkers that identify errors in text and interference checkers that identify errors in layout (spatially). In the first case, the conflict is between the written text and English orthography and in the second case the conflict is between physical objects in a spatial context. To date, the resolution process for conflicts is handled by humans. Humans make decisions to resolve conflicts just as they do to select processes, etc.

The design process controller is a critical component of a design decision support environment. It seems reasonable that any rational approach to implementation of engineering design decision support must provide the means to modify the state of the design as represented in a product/process database.

In a sequential design process, the decision structure is relatively fixed and not much thought is given to what data are needed, how design changes are propagated, or what design data might need to be updated as a result - "we just do things the same way we did on the last one". Quickly adapting product designs to evolving technologies and market conditions requires a much more flexible decision making process.

Research is needed into how information frameworks can be integrated with the tools used to create the design so that key pieces of the design decision-making process can be captured as the design definition is created. Current information frameworks require an "extra step" between the generation of design definition and design rationale. Thus, creation of a design rationale document would require additional proposal preparation time and expense.

Emerging information framework technologies, such as object-centered systems and blackboard architectures for knowledge-based systems, are promising. The question is, can they be scaled up to "real world" design applications in terms of information quantities, heterogeneous computing environments, and response times without losing their flexibility.

7.3.4 User Interfaces

A number of effective tools exist for the development of user interfaces. For example, approaches such as forms, menus, and icons are available to ease the interface between the decision-maker and the data. In addition, recent developments in window management, interactive graphics, and query languages have made possible an entirely new level of versatility in user interfaces for high end personal computers and workstations. User interfaces currently in use (in addition to forms, menus and icons) include outliners, high level query languages, and graphics.

The above features indicate the existence of a complicated modelling and parameterization activity to set up and monitor system activity. In particular:

1. legacy systems must be modelled.
2. decision variables must be defined and mapped over application data bases.

3. decision objects must be defined.
4. decision objects mu st be inspected, updated.

The engineering decision support system thus requires the same user interfaces as a typical DBMS, but is likely to require some customized icons and actions. This includes "visual/graphic" representations of complex objects and the ability to (1) traverse a network of such objects and (2) click and explode a given object into a more detailed representation.

7.4 A Concluding Viewpoint: Why the Lack of Focus on IFT ?

The research issues identified above represent a reasonably close match to the IFT emphasis of the workshop. However, when specific research questions were identified, the focus broadened considerably, to include a wide range of fundamental questions concerning the design process. These points of view are included in Appendix D following this chapter.

It is believed that the expansion of the focus of specific research questions concerning design decision-making beyond the scope of IFT reflects the broad range of critical connections between engineering design research and computing technology that are relevant to decision making in engineering design. The application of computing technology to engineering design has produced fundamental, structural changes in the way products and processes are developed.

Progress in research on computing technology in engineering design requires a multidisciplinary approach. Using a "requirements definition" approach to involving the engineers, computing scientists may develop tools based on a superficial assessment of what goes on in the process of engineering design. This is a consequence of the fact that the engineers may not accurately assess the impact of proposed computing technology on the design process when the requirements are formulated. The computing tool both makes possible a structural change in the design process and, at the same time, requires such a change in order to be effective.

Research on the application of computing technology to engineering design must be based on a more comprehensive view of the design process. Engineering design is a creative, innovative, and evolutionary process. The magnitude of the problem is illustrated by the prevalence of the idea that modelling the "information flow" in an existing (or imagined) engineering design process can, by itself, provide insights that will lead to better engineered products, or better integration of engineering design with manufacturing and support.

If this idea has any validity at all, it is only in an extremely limited context that is much smaller than its currently advertised realm of application. For systems that must conform to laws of nature (including most engineered products having some non-software component), design decisions imply some physical realization. These physical constraints are always at least implicitly present in the structure of the design process, and often appear explicitly.

In fact, the structure of the design process is based primarily upon reasoning about these constraints, using the methods of engineering science. Thus, flows of entropy, energy, momentum, matter and information in the system being designed are implied by the physical realization. The constraints implied by these flows are critical to the structure of the engineering decision-making process. Engineering theories representing the physical realization could, in principle, be included in the information model of the design process. However, the engineering design process has other information-modelling aspects. Engineering theories are often specific to the design concept. Construction of these theories is also an integral part of the design process. The decision as to which engineering theory (or theories) to accept is an important element of the decision-making process. Thus, while the basic concept of information modelling of the engineering design process is sound, the scope of the information typically included in such an analysis is often far from adequate.

Information modelling is a basic tool of information framework technology. Thus, it would seem that more needs to be done to develop information models capable of accurately representing the complexities of the engineering design process. Addressing this issue in a satisfactory way would involve another workshop, probably larger in scope than the present one. However, with the issue of representation unresolved, a meaningful starting-point for discussions on the application of IFT to decision support was difficult to identify.

In retrospect, there is the question whether decision-making, which is fundamental to engineering design, can be meaningfully addressed within the scope of a workshop on information framework technology, as a subset of computing technology. Information framework technology (IFT) is primarily concerned with the organization, storage, and integrity of information over time. Accomplishing this efficiently, while at the same time maintaining the flexibility needed for creative engineering work, seems to be impossible without compromise. Clearly, the group would not have attempted (and did not attempt) to include "conceptual design synthesis and innovation" as an area for application of information framework technology. The requirement for flexibility in a computing environment for creating design concepts is self-evident.

When one considers established ("routine") design processes, one may form the impression that the decision-making process is highly structured, and that flexibility is not a key issue. This point of view becomes untenable if one is to make decisions concerning innovative design ideas. In an innovative design process, decision-making is only one step removed from creation of the design concept. When one deals with new technologies, requirements, or concurrencies, the design decision-making process must evolve. Thus, engineering decision-makers need the flexibility to create decision-making processes, conforming to possible physical realizations, on an ad-hoc basis. Flexibility in the computing environment is evidently prerequisite to the creation of these decision processes. While capturing the decisions in an information framework is of equal importance, exploration of the IFT issues must come after investigation of the issues involved in the creation of a decision-making process. Otherwise, one is unsure about what kind of information is being placed into the framework.

7.5 Summary of Research Needs

Several valuable ideas on engineering decision support and information framework technology (IFT) resulted from the workshop, and research priorities are given in Section 7.3. The scope of the problem of making design decisions, however, is far too broad to be fully addressed in the context of IFT alone. Fundamental assumptions concerning the environment for conceptual design innovation and synthesis, and the identification of tools needed to structure a decision-making process based on the content of a design concept, underlie each of the viewpoints expressed in this chapter. The focus of this workshop on IFT provided a different view of design decision making but it was not sufficiently broad to fully and explicitly address these assumptions. Such assumptions seem in many cases to be more closely related to views of the design process than to research issues of special relevance to IFT. To fully bring out these issues and to relate them to IFT in particular and computing technology in general, it is recommended a second workshop be held having a broad emphasis on the design process, and a secondary emphasis on the role of computing technology. Thus this present workshop shows the magnitude of the research issues that need to be addressed and the need for a broad range of research to be carried out.

8. SUMMARY

This report has summarized the findings of a workshop with industry, university and government representatives to help identify critical research issues in engineering information technology which need solution to support the design of complex engineering products into the twenty-first century. The workshop conclusions emphasize the importance of research in this area and provided numerous detailed examples of research directions. It also showed that progress to date has very limited and only a small percentage of needed capabilities is available for integrating technical information. Recommended research was organized according to the four categories.

*Engineering products and process description

*Engineering information dynamics and data models

*Very high level languages and user interfaces.

*Engineering decision support systems

It is believed that this workshop report provides the details for a national research agenda of critical importance to improvements in industrial productivity to enhance in product quality and capability, reduce development times, reduce costs and provide improve capabilities for society.

APPENDIX A - WORKSHOP PARTICIPANTS

Robert Smith, DTS 68
Transportation Service Center
Cambridge, Mass 02142
(617) 494-2654

Jim Rogers
M/S 246
NASA Langley Research Center
Hampton, VA 23665
804 864-2810

Peter Brown
National Institute for Standard Tech
Building 220
A-127
Gaithersburg, MD 20899
301 975-3513

Tony Woo
Division of Design, Manufacturing &
Computer Integrated Engineering
National Science Foundation
1800 G Street, N.W.
Washington DC 20550
202 357-7676

T. N. Bernstein
IDS Program Manager
AFWAL/FIBCC
Wright-Patterson AFB, OH 45433
513 255-9729

Rob Rucker
CIMS System
Research Center, ERC 542
Arizona State University
Tempe, AZ 85287
602 965-3709

Sarosh Talukdar
Design Research Laboratory
Carnegie Mellion University
Pittsburgh, PA 15213
412 268-8778

Daniel Schrage
Dept. of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-6257

William J. Rasdorf
North Carolina State
Department of Civil Engineering
P.O. Box 7903
Raleigh, North Carolina 27695
919 737-2331
FAX 919 737-2463

Chris Hoffmann
Purdue University
Dept. of Computer Science
West Lafayette, Indiana 47907
317 494-5185

George M. Kaler
General Dynamics/Forth Worth
MS-2208
Box 748
Fort Worth, TX 76101
817 763-1642
FAX 817 777-2115

Kevin Weiler
Ardent Computer
880 West Maude
Sunnyvale, CA 94086
408 732-0400

Abe Hersh
ICAD
1000 Massachusetts Ave.
Cambridge, Mass. 02138
617 868-2800

Art Goldschmidt
IBM/M.S. 936
Kingston, NY 12401
914 385-0240

Gopal Krishnan
Itegraph Corp.
One Madison Industrial Park
Huntsville, AL 35807
205 772-1556

James Duhig
Director-CALS
Lockheed Aeronautical Systems Co.
Dept. 72-01, Zone 6
Marietta, GA 30063
404 494-2626

Michael Wozny
Director
Rensselaer Design Research Center
Rensselaer Polytechnic Institute
Troy, New York 12180-3590
518 276-6751
FAX 518 276-2732

Larry J. Leifer
Department of Mechanical Engineering
Stanford University
Palo Alto, California 94305
415 723-1869

Charles M. Eastman
University of California-Los Angeles
Los Angeles, California 90024
213 825-1174

Z. Bzymek
Department of Mechanical Engineering
University of Connitecut
Storrs, Conn. 06269
203 486-2275

Shan Navathe
Data Systems Res. & Develop. Center
E470 Computer Science & Engng. Bldg.
University of Florida
Gainesville, FL 32611
904 335-8456

Pete Whelan
Motorola
CIMS
Georgia Institute of Technology
Atlanta, GA 30332
404 894-5639

Mark Fourman
Ontologic
47 Manning Rd.
Billerica, MA 01821
617 667-2383

Sandra Heiler
Xerox - AIT
4 Cambridge Center
Cambridge, MA 02142
617 868-4700

Robert Fulton
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-7409
FAX 404 894-8336

Edward Omiecinski
Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332
404 894-3160

Georges Fadel
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-8168

Jon Colton
School of Mechanical Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-7407

James Craig
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-3042

Ed Rogan
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332
404 894-2000

APPENDIX B

INFORMATION FRAMEWORK TECHNOLOGY FOR INTEGRATED DESIGN/ENGINEERING SYSTEM WORKSHOP

March 14-15, 1989
Callaway Gardens, Georgia

AGENDA

March 13, 1989

7:00-9:00 pm	Welcoming Reception	

March 14, 1989

7:30 am	Continental Breakfast	
8:30 am	Welcome and Plans	R. E. Fulton and J. I. Craig (Ga. Tech)
9:00 am	Opening Remarks	Tony Woo, (NSF)
9:25 am	Keynote Address: Design as an Information Driven Process	M. J. Wozny, Director Rensselaer Design Research Center (Former NSF Director - Design, Manufacturing and CIE)
10:00 am	Break	
10:15 am	Critical Issues in Engineering Information Management George Kaler (General Dynamics), Daniel Schrage (Georgia Tech), Charles Eastman (Univ. of California), Lawrence Rosenfeld (ICAD)	
12:00 Noon	Lunch	
1:00 pm	Breakout Discussions Topics and Leaders **Engineering Product and Process Description** George Kaler (General Dynamics) Jim Duhig (Lockheed) **Engineering Information Dynamics and Data Models** Shan Navathe (University of Florida)	

	Very High Level Languages and User Interfaces
	Michael Wozny (Rensselaer Polytechnic Inst.)
	Charles Eastment (Univ. of California - Los Angeles)
	Engineering Decision Support Systems
	William Rasdorf (North Carolina State)
	Ed Rogan (Georgia Institute of Technology)
3:15 pm	Break
3:30 pm	Breakout Discussions
5:00	Adjourn
6:00 pm	Cash Bar
6:30 pm	Dinner
8:00 pm	Group Reporting/Discussions
10:00 pm	Adjourn

March 15, 1989

7:30 am	Continental Breakfast
8:30 am	Reconvene
8:30 am	Breakout Discussion and Report Preparation
12:00 Noon	Lunch
1:00 pm	Group Final Reports and Discussion
3:30 pm	Adjourn
5:00 pm	Arrival at Atlanta Airport

APPENDIX C

CALS OVERVIEW

In September 1985, the Deputy of Secretary of Defense issued a statement to the Secretaries of the military departments approving recommendations of a DoD-Industry Task Force on Computer-Aided Acquisition and Logistics Support (CALS). These recommendations were designed to achieve major improvements in supportable weapon system design, and to improve the accuracy, timeliness, and use of technical information.

A strategy was initiated to effect these improvements and to transition from the current paper-intensive weapon system design, manufacture, and support process to a largely automated and integrated mode of operation. The DoD Components were directed to establish plans to acquire, process, and use technical information in digital form. Major weapon system new starts, development, and modification programs were to begin to develop their acquisition strategies.

In August 1988, the Deputy Secretary of Defense issued another memorandum to the Secretaries of the military departments stating that major steps had been taken towards routine contractual implementation of CALS (Figure C.1) throughout DoD and industry. The memorandum ***upheld the issuance of CALS military standards for digital data delivery and access, and required options for access to, or delivery of, technical data in digital form for weapon systems entering development in Fiscal Year 1989 and beyond.***

In response to the ever increasing flow of paper generated by the weapon system acquisition process, both DoD and industry are investing in the automation of a variety of functional areas to improve overall productivity and quality. The lack of integration makes it hard to design systems "right the first time" and leads to costly design changes to ensure producibility and supportability. Currently, many different incompatible automated systems are used by weapon system contractors to enter, update, manage, and retrieve data from weapon system data bases. In many cases, information that is ultimately delivered to the government is created from disparate data bases, reduced to paper, and then often re-entered into government data bases.

Both the amount of paper, as well as the great potential for errors due to uncontrolled duplicate data has created the need to improve the weapon system acquisition and support processes.

CALS is addressing these problems and is progressing along an evolutionary path that will result in significant improvements in the way in which DoD and industry do business. In the near term, between now and the mid 1990s, paper document transfer will be replaced by digital file exchanges. In the longer term, advanced engineering and manufacturing information technologies, integrated product data bases, and information models will be developed. These information models and data bases will be capable of including all the information needed for design, manufacture, and support of weapon systems -- making it accessible to authorized industry and DoD users through electronic means. This is to take place in the mid 1990s and beyond, with a target of fully integrated information within the weapon system life cycle processes.

The objectives of the CALS program are to improve the timeliness, reduce the cost and improve the quality of weapons systems and their supporting technical data. These objectives are addressed through five strategic thrusts of (1) standards, (2) technology development and demonstration, (3) weapons system controls and incentives, (4) DoD systems and (5) management. CALS efforts along those thrusts are outlined in Figure C.2.

Finally, it should be emphasized that the ***CALS objectives and strategic thrusts are of critical importance to the non DoD commercial sector as well.*** CALS issues can have significant beneficial impact on productivity and quality across a broad industrial environment and can significantly aide a company's competitive capability. CALS has been evolving over the past 20 years in terms of need and the technology base and is an idea whose time has arrived. The DoD CALS initiative is providing needed leadership for a focused thrust to meet a national need. CALS will push, pull, stimulate and integrate a needed capability which if left alone, may evolve aimlessly; if led, can significantly improve the U.S. industrial capability.

CALS EDUCATION AND TRAINING ISSUES

CALS is the technology, culture and methods to transition from a paper driven weapons development process to a paperless environment. As such it embodies managing technical information as a critical DoD resource. Its components include use of processes and computer technology by both industry and the government to store, retrieve, manipulate, exchange, modify, protect, update, reconfigure and otherwise manage technical data for DoD weapons systems over the life time of the systems. While computer technology and management procedures are relatively well developed for technical computations, they are primitive for managing engineering information. While specific tasks are supported by automation, the development and support of weapons systems is largely a paper driven process. The critical need for CALS has arisen in recent years due in part to the dramatic explosion of technical data associated with weapons systems. This need continues to rapidly grow as weapons systems become more complex, as development teams become more diverse, and as development times become more acute. Instituting information driven processes becomes further exacerbated because current development procedures and corporate cultures are entrenched in paper driven approaches.

Weapons system development teams within DoD and its contractors are not well trained in CALS issues and approaches. The CALS strategy is in fact a response to this need and addresses both the technical issues of a paperless product development as well as the cultural issues within the broad DoD government and industry community. In many cases the government issues are more complex than those of industry and industry will also be expected to aide the government in achieving a CALS strategy. As CALS technical and management approaches are being developed, a critical issue is the implementation of appropriate education and training programs to accommodate and prepare for CALS.

The education and training for CALS is a complex process which needs to be understood and fostered. While some progress in CALS education will take place in an evolutionary way, an agressive education program is necessary of CALS implementation is to achieve even the initial benefits. Appropriate education and training is therefore essential for the CALS strategy to achieve the possible financial benefits.

Providing an appropriate education and training program for a CALS environment is not easy and has many components. The CALS technology and methods impact the full weapons system design, manufacturing and support environment. CALS technology contains such information technology elements as

Product data models

Engineering data exchange

Engineering information technology

Life cycle design concepts

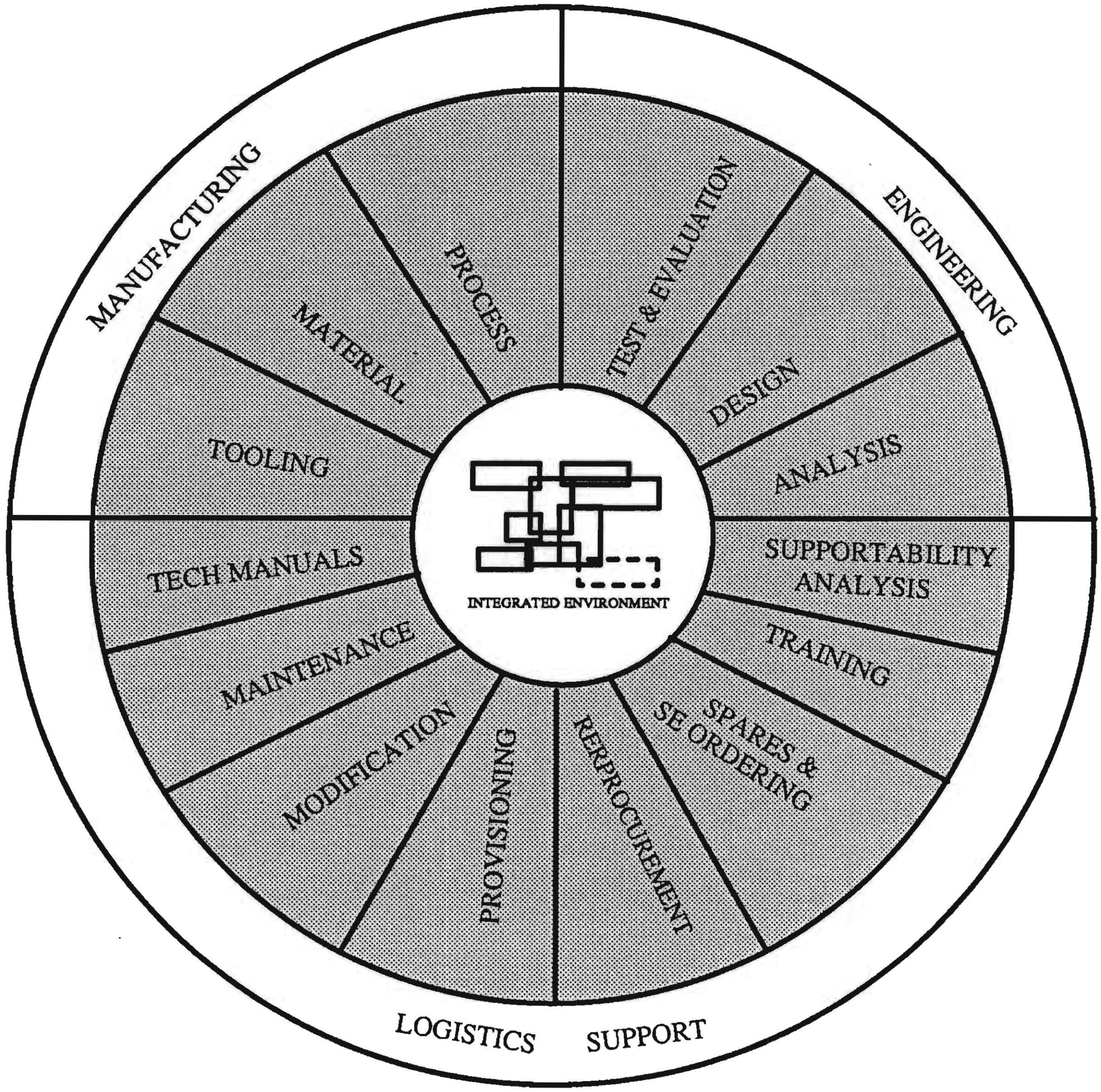
Computer based design, development and support methodology

Management of information driven processes

The CALS strategy involves significant cultural issues within DoD and industry through the effective utilization of life cycle design concepts and an information based development process. CALS implementation will require close linkages accross organization that are well entrenched at the highest levels and have a long history of work separation and control. It will require development of approaches by which decisions, control and actions are carried out based on digital database information rather than through pieces or sets of paper. It will require effective acquisition approaches through which industry can deliver and DoD can procure and use digital databases as contract deliverables rather than volumes of paper documents.

Such information based CALS technology and methods are to be used throughout the product development life cycle including design, manufacturing and support. Furthermore, these technologies and methods will not be stagnant but will continue to grow and change. Thus CALS education and training will not only have startup issues but must also be a continuing evolutionary process. In many respects CALS itself is a journey, not an endpoint. Finally, an effective education and training program cannot be limited to a classroom environment but must have ready access to computer based hardware/software system for examples, demonstrations and hands on experience.

CALS is a Dod and Industry strategy for the transition from paper-intensive acquisition and logistic processes to a highly automated and integrated mode of operation for the weapon systems of the 1990s.



CALS addresses the generation, access, management, maintenance, distribution, and use of technical data in digital form in the design, manufacture, and support of weapon systems, ships, and equipment.

Figure C.1. What is CALS?

CALS STRATEGIC THRUSTS	The CALS initiative is directed towards transition from the paper-intensive, non-integrated weapon system design, manufacturing, and support process to a highly integrated and automated mode of operation. The five main elements of the CALS strategy are outlined below.
STANDARDS	Standards are crucial to the creation of a CALS integrated environment for electronic data access and transfer. CALS will facilitate the transfer of logistic and technical information between industry and government by leveraging existing international and national standards and accelerating the development of new standards to support future requirements.
TECHNOLOGY DEVELOPMENT & DEMONSTRATION	Development and demonstration of new technologies that can support the creation, storage, and secure dissemination of a large volume of digitaized data is essential for the successful implementation of the CALS concept. CALS will support the development of integrated data base technologies that displace paper and enable redefined processes over the entire weapon system life cycle.
WEAPON SYSTEM CONTRACTS & INCENTIVES	DoD weapon system contracts with industry form the basis for implementing CALS standards and integration requirements. The CALS objective is to provide an orderly transition to a new way in which DoD and industry will do business, and to facilitate industry investment in automation and integration.
DOD SYSTEMS	Capabilities to improve readiness ultimately depend on the modernization of the DoD support infrastructure. DoD information systems must be able to receive, transmit and use digital technical data in weapon system life cycle management support activities. Current efforts include development of a corporate architecture and plan which is providing the framework for information systems modernization.
MANAGEMENT	DoD is developing the corporate plans and architecture to establish the overall direction for CALS implementation strategy. An important aspect of this strategy is to maintain a close liaison with industry and other government agencies. Army, Marine Corps, Navy, Air Force, and the Defense Logisitics Agency have prepared plans, are educating their program managers, and are proceeding with implementation.

Figure C.2. CALS Strategic Thrusts.

APPENDIX D
IFT and the Design Decision-Making Process Defining a Design Decision

* Design is typically done by interdisciplinary teams, using heterogeneous design support tools.
* Design is an iterative process that progressively refines both the requirements and the design to converge on a satisfactory product.
* An information technology based Design Decision Support System (DDSS) can contribute in a cost-effective way to improved product quality and reduced product development time.
* Design decision-making can be modeled as a process where decisions occur when the designer's attention shifts between requirements refinement and design refinement. (Refinement and exploration may be synonyms).

Definition of a design decision

How are concurrent decisions made by multiple agents? More specifically:

1. How are decision points specified/recorded/identified?
2. Can a single decision system be created (in a multidisciplinary environment)?
3. How can historical decisions be captured/managed?
4. How can design changes be propagated (among different disciplines)?
5. What kinds of support for human conflict resolution can be provided by information framework technology?
6. What design decisions are made, when are they made, and by whom are they made during a typical product evolution cycle in a particular industry?
7. What constitutes a complete and internally consistent design requirement document in a particular industry and how can it be maintained dynamically to reflect the current design state?
8. What constitutes a design decision in a particular industry and how can the record of designer activity be segmented to "time and date stamp" these decisions?

Representing Design Decisions

In order to be able to represent design decision-making, it is necessary to be able to identify the information that is needed for decision-making and representation (specification). A list of possible information required is:

a. Product description: this can be thought of as (name, value) tuples of all parameters (design variables). The objective of making any decision is to assign values to design variables.

b. Process description: a list of all steps (activities) one has to go through in order to design or manufacture a product.

c. Decision making points: these are steps within the process sequencing where decisions are to be made.

d. Requirements: The product characteristics (desired) listed by the person/organization which requested the product be designed.

e. Application/support data generator: Software/computer programs which are used to generate data to be used in a decision making point. Examples: FEM, cooling pattern generator, mechanism/kinematics model, interference checker.

f. Support data: Data generated during analysis /data available from past decisions, knowledge bases, standard catalogs, governmental regulations, etc.

g. Record of decisions made (history).

The above information must be specified in some formal way so that it can be accessed and operated on. Key elements in the representation of information are:

a. Product description

$$(x_1 = x_1^0, x_2 = x_2^0, \ldots, x_n = x_n^0),$$

where the x_i's are parameters or attributes of the design, and the x_i^0's are values, or sets of values *(which may be complicated data objects requiring specialized application programs for their interpretation).*

b. Process description

$$(s_1, s_2, \ldots, s_m)$$

c. Process sequencing

$$(s_5, s_3, s_8, s_{10}, s_3, s_{18}, \ldots, si_m)$$

d. Decision Making Points (a subset of sequencing), e.g.

$$(s_3, s_8, s_{10})$$

e. Requirements

$$(R_1, R_2, \ldots, R_n)$$

f. Application/Support data generator.

$$(A_1, A_2, \ldots, A_n)$$

g. Support data

$$(Sd_1, Sd_2, \ldots, Sd_n)$$

h. Decisions made

$$(d_1, d_2, \ldots, d_n)$$

The key research areas and research issues in design decision representation are summarized below:

a. Is the information given above complete to make design decisions ?

b. The specification of information needs to be 'robust' [in some sense].The above enumerations need to be formulated in the form of some language.

c. How to identify a "decision making point"? i.e . - a survey of the nature of the decisions made in engineering design; standardization of "processes"

d. How to represent them in the system (relational database model, object-oriented database?)

Defining Dynamics/Interrelationships of Decision Support Information.

Specific application requirements can be summarized as follows:

a. There must be a way to embed decision making points in sequencing of process description, and also decisions to be made.

b. There must be a way to tie product description (design variables) to these decision making points so that one can associate/determine what design variables get values assigned to them as a result of a decision.

c. There must be ways to specify which support data is generated by what application.

d. There must be ways to specify which application/support data generator is to be applied on the product description at what decision making steps.

e. There must be ways to specify which support data is used in a decision making point.

f. There must be ways to specify which design variables are dependent on which other design variables and in case of change of a design variable value to what step of the process is reverted warranting revisiting of an earlier decision making point.

The representation of these issues can be summarized in the same notation as used previously as follows:

a. embedding decision making points in process sequence.

$$(s_5, (s_3,d_1,d_2),(s_8,d_3,d_8), (s_{10},d_5), \ldots)$$

where (s_3,d_1,d_2) is an example of a decision point, s_3 is a process step, and d_1 is a decision to be made.

b. associating product description to decision making points

$$(s_3,x_3,x_8,x_9)$$

where in decision-making point s_3 (a process step), values for x_3,x_8, and x_9 design variables will be assigned.

c. application/support data generator to support data

$$(A_1, Sd_2 , Sd_5, Sd_6)$$

where application program A_1 generates support data Sd_2, Sd_5, Sd_6

d. application/support data generator to decision making points (mapping)

$$(s_3, A_1) \text{ -> } (Sd_5, Sd_6)$$

where for decision point s_3 use A_1 to generate Sd_5 and Sd_6.

e. use of support data in a decision making point

$$(s_3, Sd_5) \text{ -> } d_1$$
$$(s_3, Sd_6) \text{ -> } d_2$$

where for the s_3 decision making point, use support data Sd_5 to make decision d_1; also at s_3, use Sd_6 to make decision d_2.

f. design variable dependency and process re-visit

	x_1	x_2	x_3	...	x_n
x_1		s_5	s_6		
x_2	s_{18}				
x_3					
⋮					
x_n					

where an entry S_k in the i,j slot means that the design decision-making process needs to return to step S_k if design variable x_i has changed values, since it may affect the value of x_j assigned in decision making step S_k.

The key research areas and research issues for defining dynamics and interrelationships of decision support information are summarized below:

a. Are the relationships given above complete; are they adequate for decision making.

b. How are these specified formally

   i. Non-procedural language?

      - constructs

      - parser

   ii. Graphical user interface to represent states.

c. How are they stored?

   - relational database tables

- object-oriented database
- directed graphs

d. There need to be formalisms to specify what information needs to be captured in every decision making point to carry on a decision activity log, e.g.

$$(s_3, x_1, x_5, \text{<date>}, \text{<time>}, \text{<user>}, Sd_1, Sd_8, \ldots)$$

In decision making point s_3 one must log (into a file/table) the values of design variables x_1, x_5., the date and time when the decisions are made, who made the decision, and values of support data Sd_1,Sd_8, for example.

The Design Rationale and Functional Requirements for IFT

A product will typically undergo a considerable number of design changes in its life cycle. Currently, design decisions concerning the impact of these changes must often be made outside the context of the original design rationale, leading to subsequent problems.

Evaluation of a proposed product against a design requirement frequently requires that the design rationale be reconstructed. The cost and time required to make these evaluations would be reduced if the design rationale-specifically a time history of design decisions-could be delivered as part of the proposal.

Some of the key functional requirements for information technology needed in design decision support systems are outlined below:

1. The temporal relationship among design activities and decisions must be preserved.
   Why? - It has been observed that important design activity elements may be too small to support post facto interpretation. However, seen in temporal context these pieces comprise whole and important factors.
2. The decision support system must be an integral part of the design environment.
   Why? - Many "decisions" are not meaningful outside the context of the design. And, during design review, it is crucial to see these decisions in context for they are vital pieces of information for extracting the rationale behind patterns of decisions.
3. The decision support system must preserve parametric relationships.

Why? - while decisions are typically reported as binary (yes/no) events, they are in fact based on fuzzy estimates of the impact of a specific variable or strategy on the net state of the design. It is important that the decision can be revisited under alternative conditions and its sensitivity tested.

4. The decision support system must support simultaneous examination of design alternatives.

   Why? - In support of "trade-off analyses", the design decision support system must allow parallel "viewing " and "browsing" through alternative design histories. Specific decisions may have to be attached to multiple bindings in the design data base.

5. The decision support system must support and encourage extension by the designer and design manager.

   Why? - Early experience with the generation and conservation of design knowledge indicates that "personalization" of the design environment by the designer provides important clues to the design rationale and strongly supports designer productivity. By inference, the decision support system will perform better if this function is supported.

6. The design environment and associated decision support system must support automated derivation of design rationale.

   Why? - Early experience with design knowledge capture tools suggests that both the designer and the reviewer benefit significantly from re-running past design activities to re-examine lines of thinking. In time, these design records my be subject to automated rationale extraction. Such a capability would contribute importantly to design validation, constraint management and liability control.

Global and Discipline Decision-Making

A step-by-step design decision-making process can be organized as follows:

(1) understand the requirements

(2) define the information (design data) needed to make the decision

(3) determine the data dependencies

(4) determine when decisions should be made

(5) collect the design data

(6) present/review design data

(7) make the decisions

Decisions are to be made both at the global design level and at the disciplinary design level.

The types of information (data) needed to make decisions must be determined. One way making this determination is to examine the dependencies of the constraint and objective functions on the various design variables. These dependencies are to be collected and stored in a global database and disciplinary databases.

Information about these dependencies can be used to integrate different engineering disciplines. To integrate the decision-making processes across multiple engineering design and analysis disciplines, data output from any discipline must be expeditiously made available to other disciplines. Where the information is needed can also be determined by the dependencies.

To make a decision as to the best step to take to a new design state from the global or system viewpoint (as opposed to discipline or subsystem objectives), the effect of a design variable change, proposed in one discipline, on all other disciplines, and on the system as a whole, must promptly be made known to the decision makers (to at least a first order approximation). This implies that a sensitivity analysis, both system and disciplinary, must be coupled with the system (global) and disciplinary databases to retrieve and process the correct data in order to provide information to the decision makers. The alternatives and tradeoffs from the changes proposed by the various disciplines must be evaluated to determine the best choices, and decisions must be made to resolve conflicts and determine what is best from the system standpoint.

Decisions must also be made at the discipline level. The data and decisions are similar to those described above for the system. Effects from the system level, and changes from other disciplines on which a discipline is dependent must be considered before a change is made.

Finally, research is also needed to support engineers in determining when decisions should be made. When are changes sufficient to force a change? What disciplines need to have processes executed as a result of changes to the global system or another discipline?